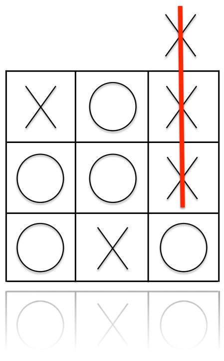
STEFAN LEIJNEN

CREATIVITY & CONSTRAINT

IN ARTIFICIAL SYSTEMS



Creativity and Constraint in Artificial Systems

Proefschrift

ter verkrijging van de graad van doctor
aan de Radboud Universiteit Nijmegen
op gezag van de rector magnificus prof. dr. Th.L.M. Engelen,
volgens besluit van het college van decanen
in het openbaar te verdedigen op woensdag 3 december 2014
om 16.30 uur precies

door

Stefan Leijnen

geboren op 29 maart 1982
te Eindhoven

Promotoren:

Prof. dr. T.M. Heskes

Prof. dr. T.W. Deacon (University of California, Berkeley, USA)

Manuscriptcommissie:

Prof. mr. dr. M. Hildebrandt

Prof. dr. J-J.Ch. Meyer (Universiteit Utrecht)

Prof. dr. S. Colton (Goldsmiths College, University of London, UK)

Printed by BoxPress, 's-Hertogenbosch

Cover & layout: Stefan Leijnen

Leijnen, S. (2014) Creativity and Constraint in Artificial Systems. PhD thesis, Radboud University Nijmegen. ISBN 978-90-8891-993-0

CONTENTS

Contents v

Preface ix

1 Introduction 1

1.1 Science and Philosophy of Creativity 2
1.2 Artificial Creativity 5
1.3 Creativity & Constraint 8
1.4 Research Questions 10
1.5 Methodology 11
1.6 Outline of the Thesis 12

I Creativity in an Artificial Agent Society 15

2 Creation and Imitation in a Multi-Agent System 17

2.1 Introduction 17
2.2 The Modeling Approach 19
2.3 Architecture 21
2.4 Implementation 25
2.5 A Typical Run 26

3 Cultural Evolution in EVOC **27**
3.1 Proportion of Creators to Imitators 28
3.2 Innovation Rate 35
3.3 Time Series Analysis 37
3.4 Creative Leadership 43

II Creativity and Constraint **51**

4 Creating Symbols **53**
4.1 The Meanings of Symbol 53
4.2 Simulated Learning 56
4.3 Experiments 58
4.4 Conclusions 60
4.5 Discussion 62

5 Computation, Creativity and Constraint **65**
5.1 Introduction 65
5.2 Combination, Exploration and Transformation 66
5.3 Self-Programming Constraint 67
5.4 Conclusions 68

6 Constraint, Self-Organization and Autogenesis **69**
6.1 Introduction 70
6.2 Autogenesis 71
6.3 The Autogenic Automaton 76
6.4 Constraint Generation 79
6.5 Constraint Elimination 83
6.6 Constraint Preservation 84
6.7 Constraint Selection 88
6.8 Conclusions 90
6.9 Appendix: Model Description 91

III Creativity Support Tools **95**

7 Chronicling Cultural Ancestry through Conceptual Classification **97**
7.1 Introduction . . . 98
7.2 Phylogenetic Approaches . . . 98
7.3 Lipo's Network (LN) Approach . . . 102
7.4 The Conceptual Network (CN) Approach . . . 106
7.5 Results . . . 109
7.6 Discussion . . . 111

8 Creativity in Procedural Content Generation for Video Games **113**
8.1 Introduction . . . 113
8.2 Designing for Creativity . . . 114
8.3 Experiments . . . 116
8.4 Tailoring PCG Tools . . . 122
8.5 Conclusions . . . 124

Epilogue **127**

List of Publications **133**

Bibliography **135**

Curriculum Vitae **143**

Samenvatting **145**

Summary **147**

PREFACE

I shall be telling this with a sigh
Somewhere ages and ages hence:
Two roads diverged in a wood, and I –
I took the one less travelled by,
And that has made all the difference.

from *The Road Not Taken*, Robert Frost (1916)

They say that writing a dissertation is a journey along a lonely road. When travelling alone, with no one holding you back, you are more likely to head far into unknown territory. However, as you get lost, there is no one you can turn to for help. You are out there on your own.

It can be like that, at times, when you choose the road less travelled by. Yet, whenever I turned for help, I found people around me who would. At those dark moments when I was ready to give up writing a dissertation, there were people around me who convinced me to continue. When I was in need of critical reflection, there were people around

me who gave their sometimes harsh but always honest opinions. And when, at other times, I proclaimed the exceptional genius of my work after a cryptic fifteen-minute monologue that nobody could possibly follow, there were people around me who smiled patiently.

To Tom, my supervisor, who did all of the above. You welcomed me into your group, offering me academic shelter despite the eccentricity of my research topic. I thoroughly enjoyed our meetings, and I cannot adequately express how thankful I am for the opportunity you gave me, and how much this opportunity has changed my life.

To Terry, my mentor. I remember vividly how, after arriving in Berkeley for the first time, I was blown away by the audacity of your investigations. Now, almost nine years later, it remains my main source of inspiration.

To John-Jules, who stimulated me to pursue a PhD. You have been there when I needed you on every step of the way, for the past thirteen years; I cannot imagine a better professor. To Liane, you accompanied me on the first legs of my journey, and showed how science can be more than a profession. To Jeremy, Ty, Julie, Hajime, Spyridon, Jay, Bob, Alok, Josh, Rodrigo, and the other Pirates, you have made Berkeley my academic home away from home, and showed me that science can sometimes be as beautiful and enjoyable as a good conversation.

To Mireille Hildebrandt and Simon Colton, for meticulously reviewing the original manuscript and giving many excellent comments. To Palmyre, for helping me find my way to Nijmegen, and for the many occasions we could share ideas. To Pim, for the interesting discussions and joint work. To Pieter Adriaans and Marc Slors, whose work I admire, and who were willing to take part in the opposition.

To my former and current research colleagues in Amsterdam, Nijmegen, Boston, Kelowna, Berkeley, Liverpool and Utrecht. In particular to Anders Bouwer, Tom Claassen, Daniel Dennett, Joris Dormans, Stephan van der Feest, Tiha von Ghyczy, Jelle Herold, Daniel Karavolos, Carl Lipo, Nicole Messink, Joris Mooij, Apara Ranjan, Kees Rijsenbrij, Riemer van Rozen, Janos Sarbo, John Searle, Manuela Ungureanu, and Tomas Veloz. To the excellent colleagues at the game development department, for the enthusiasm, support, and fun you bring to the workplace every day.

To the friends who walked beside me on this remarkable journey, and made it far from lonely. In particular to Lars Andeweg, Lieke Bartelds, Willem-Bart Bartels, Wouter van Berkel, Lars Blaauwbroek, Arno Bode, Lubbert de Boer, Paul Brinkkemper, Michel Crevecoeur, Martin Deen, Marcel van Dijk, Alex Drapers, Tim van Erven, Janneke van Es, Wouter le Fèvre, Rene Gelens, Joost van der Hagen, my Hardslag teammates, Jasper 't Hoen, Cornelis Huizinga, Merlijn Hurx, the staff of Café Marktzicht, Martijn Huttenhuis, Geert Jonker, Wouter van Keulen, Stijn Kieft, Matthijs Kloosterboer, Bart Koopman, Vinod & Bharati Mittal, Eric Lai, Sieuwert van Otterloo, the eternal legends of Pangster's Paradise, Rutger Rietveld, Eric Roefs, Eric Sillekens, Bram Snel, Sophia, Garry Sotnik, Eduard Storm, Erik Tillema, Jan 'Tjibbe' Veenstra, Remon te Velthuis, Daan Vrieler, Silvain van Weers, Erik 'Wobbe' Wesselius, and Mark Westerbeek.

To my parents, Puck and Rina, for your encouragement in every phase of my life.

To Natalie, for the way you see the good in everyone and everything.

To Emma, for all the love and happiness you bring to my life, and for supporting me these years past. I could not have done this without you.

Thank you.

Stefan Leijnen, Utrecht
October 10, 2014

In the beginner's mind there are many possibilities, in the expert's mind there are few.

Shunryu Suzuki, 1970

1

INTRODUCTION

The word *creativity* has held different meanings and connotations over time. Tracing back its etymological roots to ancient India, Greece, and Rome, *to create* means "to make, produce" (Sanskrit *kriya*; Greek *ktidzo*; Latin *creare*) and is related to "to arise" and "to grow" (Latin *crescere*). Creation, it was thought, involved the production of artifacts in a physical sense – with observable results – through imitation or emulation. Cognitive activities such as discovery, insight and guessing formed a separate domain, in the same sense that private activities do not fall within the public domain [71].

Plato regarded learning as an act of remembrance (anamnesis) with the teacher acting as a midwife for knowledge, rather than an instructor [1]. The origins of cognitive activity were attributed to divine *nous*; a theological connotation that was carried over to the Middle Ages, where creation *ex nihilo* was explained by divine inspiration, a meaning reflected today by the word 'creationism' that arose around 1880 as a reaction to Darwinism.

During the humanistic revolution of the Renaissance, and later, the Enlightenment, the causal locus of creativity shifted from the immaterial to the physical domain. *Imagination* as the origin of ideas bridged the dualistic gap between physical and cognitive production, made famous by Descartes' "Cogito ergo sum". Some early 20th century scientists would report on their creative process through introspection [65]. One often mentioned introspective report has German organic chemist August Kekulé accounting for his invention of the benzene ring:

> *"I was sitting writing at my textbook but the work did not progress; my thoughts were elsewhere. I turned my chair to the fire and dozed. Again the atoms were gamboling before my eyes. This time the smaller groups kept modestly in the background. My mental eye, rendered more acute by the repeated visions of the kind, could now distinguish larger structures of manifold confirmation: long rows, sometimes more closely fitted together all twining and twisting in snake like motion. But look! What was that? One of the snakes had seized hold of its own tail, and the form whirled mockingly before my eyes. As if by a flash of lightning I awoke."* [135]

In this succinct account, several aspects of creativity are brought to light. Kekulé describes how his insight finally emerged when his thoughts diverged away from the problem at hand ('my thoughts were elsewhere') after a period of preparation ('again the atoms'; 'repeated visions of the kind'). He also describes how cross-domain transfer helped him conceive his idea: the flames are associated with twisting snakes that are, in turn, associated with configurations of atoms. And finally, the 'flash of lightning' refers to the sudden insight he acquired in a relatively short timespan.

1.1 Science and Philosophy of Creativity

Early in the 20th century, as psychology grew into an academic discipline, imagination, preparation and other aspects of the creative process became subject to scientific scrutiny. Introspective accounts of scientists' and artists' creative processes were collected and analyzed, culminating in a general theory of creativity that consisted of five subsequent stages [164]:

1. **preparation**: knowledge of the problem domain is acquired
2. **incubation**: acquired knowledge is (unconsciously) internalized
3. **intimation**: new insights indicate that the problem may be solved soon
4. **illumination**: a possible solution emerges
5. **verification**: the solution is validated by applying it

In Wallas' five stage model, creativity is equated with scientific discovery – and understandably so, considering the inspiration for his model. Yet, creativity "[...] a particular form of taxation, a particular style of painting or dancing, a way of building a bridge or skinning a cat, a millinery design [...]" [14]. It can be argued that, indeed, some level of creativity is necessary for many everyday activities. How can people write grocery lists without using their imagination? Shakespeare is believed to have added over 1700 new words to the English language, but someone engaging in a casual conversation is also constructing new sentences on the fly, some of which may never have been uttered before. Both eminent grandmasters and less talented chess players had to learn the rules of the game, although they may use a different level of creativity to use them.

To distinguish between the creative outburst of recognized geniuses like Kekulé, Shakespeare, Mozart and Picasso [144] and everyday creativity [131], the terms "big C creativity" and "small c creativity" [94] are often used. These labels are sometimes difficult to apply. For example, it cannot be ruled out that Albert Einstein ever wrote a grocery list, and in doing so demonstrate a capacity for small c creativity. Vincent van Gogh only became recognized as an eminent artist after his death – inspiring hope for later artists that their presumed eminence will be recognized after their lifetime as well. The phlogiston theory, providing a scientific explanation for combustion, was thought to be an eminent discovery in the 17th century, only to become a prototype example of science theories proved false since its 18th century demise instigated by Antoine Lavoisier's experiments. However, despite these and similar difficulties in distinguishing eminent from everyday creativity, psychometric and psychosocial research [72, 143] shows that the origins of genius and the fruits of its labor can be categorized and studied scientifically.

A related distinction can be made between creative acts that are historically new to society (H-creativity) and those that are merely new on a personal level (P-creativity) [15]. From this definition it follows that all H-creative acts are P-creative, but not all P-creative acts are H-creative. Here, too, the distinction between individual and society is more troublesome than it may appear at first sight, as the difference between individuals and the societies dissolves when we regard people as the fabric of society [49].

In these definitions, a pivotal aspect of creativity presents itself. In considering the relation between H-creativity and P-creativity, a hierarchical shift is necessary from the level of the individual to the level of a society that is composed of individuals. A similar hierarchical shift is possible between individuals and the cells and molecules they are composed of; at this biochemical level, it would be intuitively undesirable to attribute creativity to, say, individual neurons. Unless some minimum requirement for creative acts is adopted, we risk confusing creativity with *pancreativity* - the idea that every event is principally unique, new, and therefore produced by a creative act. Therefore, any scientific theory of creativity needs to adequately explain why atoms, molecules and neurons are not creative, while humans and societies are. This fundamental idea is captured in the distinction between unintended, purposeful, and original creative acts [77] (see figure 1.1).

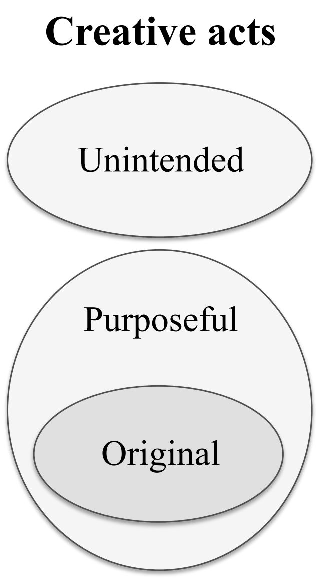


Figure 1.1: Venn diagram of three classes of acts. Unintended acts lack purpose, while purposeful acts may or may not be original (i.e. new to a group), roughly corresponding to H-creativity and P-creativity.

What separates purposeful and original acts from new but unintended acts is their value and intelligibility [76] to the actor, or their compliance with an appropriateness criterion [151]. Extrinsic independence (i.e. from environmental causation) and intrinsic independence (i.e. spontaneity) is required for purposeful novelty creation [99]. For example, it would be a trivial exercise to make a computer print Marcel Proust's "A la recherche du temps perdu" on a display – it would constitute an unintended creative act that lacks intrinsic and extrinsic independence. Considering the mechanical computational process in isolation, the program's execution lacks spontaneity; when we consider the human programmer to take part in the creative process to allow for spontaneity, the program itself is no longer extrinsically independent.

Purpose is embedded in every intended creative act. Goals, solutions, or (aesthetic) values are a systemic aspect of every intended act of creation. The requirements for purposeful creative acts can therefore be expressed in the following terms [15, 77, 120, 151]:

- **novelty**: not existing previous to the creative act.
- **usefulness**: being appropriate to the creative system's goals, problems, or values;
- **surprise**: the recognition that an act is both new and useful.

The ex nihilo production of observable artifacts, thought to arise from divine inspiration many centuries ago, is re-presented here as purposeful novelty creation; the creator recognizing both the newness and usefulness of its creative act. Many theories of creativity reflect these minimum requirements in one way or another, as they explain novelty as a combination of previously unconnected ideas through bisociation [96] or conceptual blending [47], usefulness as a fitness function over a set of random ideas [22, 145], or surprise as learning optimization [24, 52, 138].

1.2 Artificial Creativity

The invention of the universal computer, first as a mechanical and later as an electronic device, introduced an entirely new experimental facility for scientific research: a theoretical model could now be put to the test by means of simulation and comparing the calculated outcomes to what the model predicted. Many would argue that computers are capable of typically human traits such as intelligent behavior, language use, and creativity [160], cf. [139]. Or that, if not autonomously capable of human-like behavior, they can at least assist us in these domains. Chess programs, for example, exhibit all of these three types: they can autonomously play against human players, suggest moves to human players, or provide insight into the cognitive efforts involved when people play chess.

With respect to creativity, Lady Ada Lovelace, daughter of the poet Lord Byron and commonly believed to be the first computer programmer, wrote in 1843: "The Analytical Engine has no pretensions whatever to originate any thing. It can do whatever we know how to order it to perform. It can follow analysis; but it has no power of anticipating any analytical relations or truths. Its province is to assist us in making available what we are already acquainted with." [101]. This is a remarkable passage considering how, in the early days of AI, artificial intelligence and artificial creativity were almost synonymous [18, 116]. The proposal for the 1956 Darthmouth conference which marks the birth of artificial intelligence as a research field contains the following excerpt:

> ***7. Randomness and Creativity***
> *A fairly attractive and yet clearly incomplete conjecture is that the difference between creative thinking and unimaginative competent thinking lies in the injection of some randomness. The randomness must be guided by intuition to be efficient. In other words, the educated guess or the hunch include controlled randomness in otherwise orderly thinking.* [114]

The phrase 'randomness must be guided by intuition to be efficient' conceals a number of difficult challenges, as it loosely suggests that novelty requires randomness and refers to usefulness in terms of efficiency. Since 1956, many computer programs have been made to model novelty, usefulness, surprise, randomness, intuition, or efficiency, in one way or another. Examples include programs that produce visual art [26, 31], music [167], poetry [63], puns [12], mathematical concepts [29], and metaphors [80].

Whether or not computer programs (e.g. fig. 1.2 and fig. 1.3) ought to be considered as creative systems is a question open for debate, where the answer also depends on the method of evaluation. In an experimental setup similar to the Turing test [160], products created by artificial systems may be subjected to human evaluation [19]. To avoid the possible bias of the human observer that computers can't be creative, the process by which the product was created should be unknown (cf. [30]). Within this experimental framework, the necessary requirement for computational creativity is "the performance of tasks which, if performed by a human, would be deemed creative" [166].

A different method for evaluating the creativity of artificial systems is by observing the mechanical or emergent processes that they are composed of, to define the min-

Figure 1.2: Graphic art created by AARON [26]

```
Q: What is the difference between leaves and a car?
A: One you brush and rake, the other you rush and brake.

Q: What do you call a strange market?
A: A bizarre bazaar.

Q: What kind of murderer has moral fibre?
A: A cereal killer.
```

Figure 1.3: Punning riddles created by JAPE [12].

imal process requirements for creativity. Creative programs often generate, evaluate and select artifacts; some execute this task autonomously, while others require human interaction to account for the necessary 'guided intuition', or purpose, in their creative process. While the interplay between computers and humans provides an interesting perspective on their joint creative process – some computer artists even express a desire to extend interactivity and include the audience's experiences in the creative loop (Frieder Nake, pers. com.) – the systemic aspects of autonomous artificial creativity, and the scientific and philosophical implications of this possibility, are considered in the next section.

1.3 Creativity & Constraint

Consider the following puzzle: nine dots are drawn on a piece of paper as shown in figure 1.4. You are challenged to connect these dots by drawing four straight lines, without lifting your pencil from the paper.

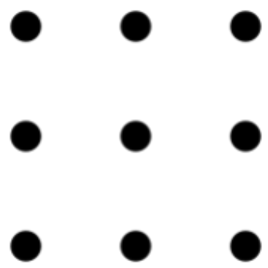


Figure 1.4: The nine dots puzzle.

A solution is presented in figure 1.5. The nine dots puzzle originally appeared as the "Christopher Columbus's egg puzzle" in the 1914 Cyclopedia of Puzzles [111] and is believed to have inspired the aphorism "thinking outside the box", which has become a common metaphor for thinking creatively. The layout of the puzzle falsely suggests a constraint that lines need to start and end at a dot, resulting in an exhaustive search for a solution that doesn't exist. However, this underlying constraint is never made explicit: it may have arisen out of convention (e.g. familiarity with similar connect-the-dots puzzles) or from Gestalt theoretical principles (e.g. perceiving the dots grouped together within a square). By extending the search space and allowing lines to be drawn outside the imagined box, a previously unimaginable solution presents itself.

Thinking outside the box is reflected in a more general sense by the realization of implicit assumptions that allows for extending or transforming the search space. The importance of this problem for learning was already pointed out by Plato in his *Meno* dialogue, as what became known as the *sophistic paradox*:

> "And how are you going to search for [the nature of virtue] when you don't know at all what it is, Socrates? Which of all the things you don't know will you set up as target for your search? And even if you actually come across it, how will you know that it is that thing which you don't know?" [126]

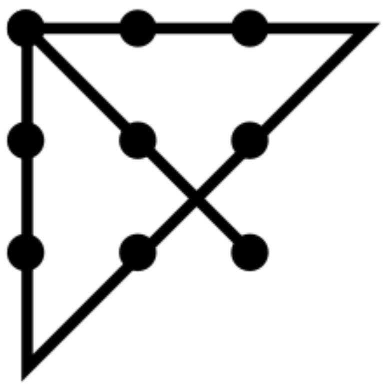


Figure 1.5: A solution to the nine dots puzzle.

In addressing the absence of evidence for Iraqi weapons of mass destruction in 2002, United States Secretary of Defense Donald Rumsfeld rephrased Plato's concern as follows:

> "As we know, there are known knowns. There are things we know we know. We also know there are known unknowns. That is to say, we know there are some things we do not know. But there are also unknown unknowns, the ones we don't know we don't know." [140]

Solving the nine dots puzzle requires venturing into the unknown unknown, i.e. to explore the possibility of alternative search spaces. This meta-search problem is often represented by three distinct types of psychological processes – or levels of surprise [15]:

- **Combinatorial creativity** occurs when novelty arises out of unfamiliar combinations of familiar ideas;
- **Exploratory creativity** makes new ideas through existing conventions;
- **Transformational creativity** alters the conceptual space itself, such that new ideas are generated that do not fit in a previous style or convention.

Combinatorial and exploratory creativity can be represented computationally as randomized combination of existing concepts, and traversal through a predefined search space, respectively. Simulating transformational creativity is straightforward, however, as a distinction between the search space within the box, and the meta-search space of possible boxes is required (fig. 1.6).

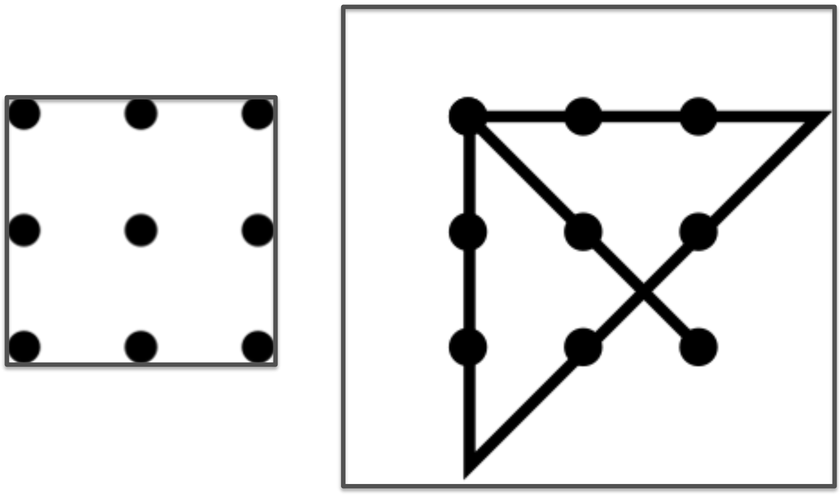


Figure 1.6: Two approaches to solving the nine-dots puzzle: lines are either drawn inside the (imagined) box formed by the dots (left), or lines are drawn outside of it as well (right). In the former case, no solution is found, whereas in the latter case, the conceptual space is not altered. With transformational creativity defined as thinking outside an established search space, neither approach complies with this transformational creativity criterion.

The possibility of eliminating a self-imposed constraint, such as the virtual box that we experience when trying (and failing) to solve this puzzle, suggests that systems may exist which transform search spaces by generating and eliminating constraints. Such systems are explored in the second part of this thesis, showing that creativity is both limited and enabled by constraint.

1.4 Research Questions

The issues discussed in the previous sections lead to the following research question:

To what extent can creative processes be modeled in artificial systems?

This question is split up into three key questions, each corresponding with one part of this thesis:

I *How can historical creativity be modeled computationally as an evolutionary process in an artificial agent society, and given this model, what parameter settings* *optimize the fitness and diversity of ideas?*

II *What are the systemic requirements for artificial transformational creativity, and how can such a system be modeled computationally?*

III *How can knowledge of creative processes be applied to the development of computational tools that support creativity?*

1.5 Methodology

The three parts differ not only in terms of what question they resolve, but also in the approach taken toward computational modeling, and toward what constitutes creativity. In the two chapters that form the first part, creativity is modeled as a group process, in order to learn how innovation and imitation impact the spread of ideas among a society of interacting agents. As such, these chapters address issues of historical creativity, though not necessarily of Big-C ideas: small, incremental steps that improve the usefulness of ideas are considered to be creative, just as giant leaps are.

Whereas in the first part, the distinction between unintended and purposeful actions is bracketed out (c.f. fig. 1.1), this distinction resurfaces in the three chapters that follow in the second part. In order to accurately model transformational creativity, the evaluation of value can no longer be delegated to an externally imposed fitness criterion; rather, this evaluation needs to form an intrinsic part of the model for the autonomous production of constraints to be transformationally creative. Computation, here, proves to be a tool rather than a model, as computer simulation allows for a non-computational system to be modeled in chapter 6.

Part III, finally, considers creativity support tools in two domains: archeology and video game development. The first chapter features a program for the automated classification of material artifacts, known as cladistics, that relies on conceptual, rather than phylogenetic, artifact attributes. This chapter, creativity features in threefold: the tool automatically creates suggestions for inheritance trees; it supports the generation of new and original models of cultural inheritance by archeologists; and third, the artifacts are themselves products of prehistoric invention. The second chapter of this part looks at tools for automated content generation for games. Considering the troublesome relation between computation and transformational creativity discussed in the second part, the applied techniques are limited to combinatorial and exploratory (rather than transformational) creativity.

1.6 Outline of the Thesis

Part I: Creativity in an Artificial Agent Society features a computer model of cultural evolution referred to as EVOC (for EVOlution of Culture), which incorporates cultural analogies of biological epistasis, mutation, and crossover of coded instructions. The benefits and drawbacks of H-creativity are analyzed at the level of agent groups, where some neural network-based agents invent new ideas through modification of existing ideas, while others merely imitate their neighbors' ideas. Although imitation appears to be a form of free-riding on the originality of others, the behavior of these imitator agents is essential as they help prevent the breaking up of co-adapted partial solutions. In **Chapter 2: Creation and Imitation in a Multi-Agent System**, the EVOC model and architecture are described, while **Chapter 3: Cultural Evolution in EVOC** presents a series of experiments investigating the optimal proportion of creative types to imitators in society, the optimal innovation rate of creatives, the effect of clustering creative agents together, time series analyses of the relation between innovator-to-imitator ratios and innovation rates, and the impact of creative leadership.

Within the EVOC framework, purpose is simulated by random variation in a neural network. **Part II: Creativity and Constraint** looks at the causal locus of novelty and usefulness, and the emergent dynamics that may ultimately be required for transformational creativity. In **Chapter 4: Creating Symbols**, the difference between indexical and symbolic interpretation is explored using a neural network simulation of chimpanzee language training experiments - crossing the symbolic threshold is perhaps the ultimate transformationally creative act. Following a series of experimental results, the systemic requirements for crossing this threshold are discussed. These requirements, and the closely related question whether computers can accurately model systems capable of symbolic interpretation, is what originally gave rise to the investigations reported in the two chapters that follow.

The limitations of computational models of emergent and hierarchical dynamics are investigated further in **Chapter 5: Computation, Creativity and Constraint**.

For a system to change its own conceptual space, it needs to be capable of producing constraints that are new and useful to itself. This requirement is discussed briefly with respect to the fixed constraints required to execute computational mechanisms.

The questions and challenges raised in these two previous chapters are then subject of the central piece of this Part, **Chapter 6: Constraint, Self-Organization and Autogenesis**. Just as EVOC frames historical creativity in terms of evolutionary biology, so is the Autogenic Automaton, a computer model of presented and discussed in this chapter, tied to a biological framework: it shows how novelty and usefulness may arise spontaneously through the abiogenic generation, elimination, preservation and selection of constraints. The chapter presents a simulation of an autogenic system, a hierarchical model of emergent dynamics that exemplifies the preservation of accumulated constraints as resulting from a reciprocal coupling between self-organizing processes. Origins of life theories often employ self-organization to account for the formative power [91] that produces life seemingly out of nothing. However, how the constraints produced by self-organization are maintained and preserved over timespans stretching long enough for evolutionary selection dynamics to occur is currently not well-understood. We show that this coupling produces a second order constraint that can resist dissipation and become replicated in new substrates over time.

Part III: Creativity Support Tools features **Chapter 7: Chronicling Cultural Ancestry through Conceptual Classification**, where a tentative program is described that suggests patterns of cultural ancestry from archeological artifacts based on their conceptual attributes (e.g. information about their function) in addition to quantitative attributes such as their size and shape. **Chapter 8: Creativity in Procedural Content Generation for Video Games** investigates how knowledge about creative processes may be applied to improve procedural content generation tools and techniques. Using a program called Ludoscope that uses transformational grammars to generate content, we find that separating generation and resolution of the proposed solutions into a dual process facilitates the design of algorithms and impacts the design of PCG tools.

PART I

CREATIVITY IN AN ARTIFICIAL AGENT SOCIETY

2

CREATION AND IMITATION IN A MULTI-AGENT SYSTEM

There are both benefits and drawbacks to creativity. In a social group it is not necessary for all members to be creative, to benefit from creativity. Some seem to merely adopt and free-ride on innovations achieved by others; in reality, however, their behavior is essential. This chapter[1] describes a simulation of cultural evolution referred to as EVOC (for EVOlution of Culture). EVOC is composed of neural network-based agents that evolve fitter ideas for actions by inventing new ideas through modification of existing ones, and imitating neighbors' ideas.

2.1 Introduction

Computer science is drawing ever more extensively upon the natural world for inspiration in the design of search algorithms, optimization tools, problem solving techniques, and even computer-based artistic expression. What nature - a most effective problem solver - has come up with is the human mind itself. The brain's effectiveness derives largely from the fact that it is endlessly creative, able to break out of ruts and come

[1]This chapter is based on [106] Leijnen, S. and Gabora, L. (2009). How creative should creators be to optimize the evolution of ideas? A computational model. *EPTCS, 9*:108–119.

up with ideas and solutions that are new, useful, and appealing. Not only are we individually creative, but we build on each other's creations such that over the centuries our ideas and inventions can be said to have evolved. In order for computer scientists to put to use the process by which creative ideas evolve through cultural exchange we must first develop better computational representations of the process. This chapter investigates one of its aspects: the interaction between how creative individuals are, and how numerous they are in a society.

Our capacity for self-expression, for finding practical solutions to problems of survival, and coming up with aesthetically pleasing objects that delight the senses, all stem from the creative power of the human mind. However, there are also considerable drawbacks to creativity. An original solution to one problem often generates other problems or unexpected negative side effects that may only become apparent after much has been invested in the original solution. Moreover, creative individuals are more emotionally unstable and prone to affective disorders such as depression and bipolar disorder, and have a higher incidence of schizophrenic tendencies, than other segments of the population [3, 50, 85, 86, 152]. They are also more prone to abuse drugs and alcohol [68, 69, 112, 118, 119, 134] as well as suicide [70]. Also, creative people often feel disconnected from others because they defy the crowd [150, 153]. Fortunately, in a group of interacting individuals, only a fraction of them need be creative for the benefits of creativity to be felt throughout the group. The rest can reap the benefits of the creators' ideas without having to withstand the dark aspects of creativity by simply copying, using, or admiring them. After all, few of us know how to build a computer, or write a symphony, or a novel, but they are nonetheless ours to use and enjoy when we please. In a society of interacting individuals capable of imitation, some members can capitalize on the benefits of creativity without incurring the drawbacks by merely imitating their creative peers. This opens up some interesting questions. In order for a culture to evolve optimally - in terms of collectively exploring the space of possible solutions - what is the ideal ratio of creators to imitators? Secondly, how creative should these 'creative types' be?

We have investigated these questions using an agent-based modeling approach. The agents are too rudimentary to suffer any of these affective penalties of creativity. We focus on two negative consequences that are much more straightforward and no less relevant. First, innovation is a slow and frustrating process. Only few innovators are lucky enough to progress quickly, the rest are struggling and straggling behind. Unless, of course, they too get a chance to imitate what their best peers have achieved and build on that. In other words, it is not only the imitators who can benefit from copying; most innovators do so too. Second, and related to the above, an excess of creative agents too fully engaged in the process of invention become effectively insulators or blockages in the rapid diffusion of the best ideas. If so, they hurt themselves just as much as their 'lesser' imitating brethren.

Whereas in the earliest versions of the computer model used here, all agents were equally capable of both inventing and imitating [53], in a subsequent version greater individualization was possible [107]. Each agent could be a pure imitator, a pure creator, or something in between. We found that for low probabilities of invention for creators, the mean fitness of ideas increased as a function of the percentage of creators in the society, but for higher invention probabilities, the optimal ratio of creators to imitators followed a nonlinear decreasing function. Thus as a general rule, the more creative the creators were, the less numerous they should be. In this chapter and the next, we report on a much more extensive investigation that employs a more detailed and sophisticated analysis of these questions.

2.2 The Modeling Approach

EVOC consists of neural network-based agents that invent ideas for actions, and imitate neighbors' actions [57]. EVOC is an elaboration of Meme and Variations, or MAV [53], the earliest computer program to model culture as an evolutionary process in its own right. MAV was inspired by the genetic algorithm (GA), a search technique that finds solutions to complex problems by generating a 'population' of candidate solutions through processes akin to mutation and recombination, selecting the best, and repeating until a satisfactory solution is found. Although MAV has inspired the incorporation of cultural phenomena (such as imitation, knowledge-based operators, and mental simulation) into evolutionary search algorithms (e.g. [98]), the goal behind MAV was not to solve search problems, but to gain insight into how ideas evolve. It used neural network-based agents that could (1) invent new ideas by modifying previously learned ones, (2) evaluate ideas, (3) implement ideas as actions, and (4) imitate ideas implemented by neighbors. Agents evolved in a cultural sense, by generating and sharing ideas for actions, but not in a biological sense; they neither died nor had offspring. The approach can thus be contrasted with computer models of the interaction between biological evolution and individual learning [8, 9, 78, 79, 84].

MAV successfully modeled how 'descent with modification' can occur in a cultural context, but it had limitations arising from the outdated methods used to program it. Moreover, although new ideas in MAV were generated making use of acquired knowledge and pattern detection, the name 'Meme and Variations' implied acceptance of the notion that cultural novelty is generated randomly, and that culture evolves through a Darwinian process operating on discrete units of culture, or 'memes'. Problems with memetics and other Darwinian approaches to culture have become increasingly apparent [17, 51, 54, 55, 56, 87]. One problem is that natural selection prohibits the passing on of acquired traits (thus you don't inherit your mother's tattoo)[1]. In culture, however, 'acquired' change - that is, modification to ideas between the time they are learned and the time they are expressed - is unavoidable. Darwinian approaches must assume that elements of culture are expressed in the same form as that in which they are acquired. Natural selection also assumes that lineages do not intermix. However, because ideas cohabit a distributed memory with a multitude of other ideas, they are constantly combining to give new ideas, and their meanings, associations, and implications are constantly revised. It has been proposed that what evolves through culture is not discrete memes or artifacts, but the internal models of the world that give rise to them [54], and they evolve not through a Darwinian process of competitive exclusion but a Lamarckian process involving exchange of innovation protocols [55, 56]. EVOC incorporates this in part by allowing agents to have multiple interacting needs, thereby fostering complex actions that fulfill multiple needs. Elsewhere [57, 58] results of experiments using different needs and/or multiple needs are described, as well as how cultural evolution is affected by affordances of the agents' world, such as world shape and size, population density, and barriers that impede information flow, and potentially erode with time. This research investigates how different proportions of creative to uncreative agents affect the fitness and diversity of ideas.

[1] That isn't to say that inheritance of acquired traits never occurs in biological evolution; it does. However, to the extent that this is the case, natural selection cannot provide an accurate model of biological evolution. Because inheritance of acquired traits is the exception in biology not the rule, natural selection still provides a roughly accurate model of biological evolution.

2.3 Architecture

This section describes the key components of the agents and the world they inhabit.

2.3.1 The World

EVOC consists of an artificial society of agents in a two-dimensional 10x10 grid-cell world. Agents are stationary, such that agents always have the same four neighbors. Although EVOC allows for sparse population (i.e. not every grid square contains an agent), the world is completely populated in the experiments reported in the next chapter.

2.3.2 The Agent

Agents consist of (1) a neural network, which encodes ideas for actions and detects trends in what constitutes a fit action using knowledge based operators, and (2) a body, which implements actions.

In each iteration, every agent has the opportunity to (1) acquire an idea for a new action, either by imitation, copying a neighbor, or by invention, creating one anew, (2) update the knowledge-based operators, and (3) implement a new action. To invent a new idea, the current action is copied to the input layer of the neural network, and this previous action is used as a basis from which to generate a new one. For each node the agent makes a probabilistic decision as to whether change will take place. If it does, the direction of change is stochastically biased by the knowledge-based operators using the activations of the SYMMETRY and MOVEMENT nodes. Mental simulation is used to determine whether the new idea has a higher fitness than the current action. If so, the agent learns and implements the action specified by the new idea. To acquire an idea through imitation, an agent randomly chooses one of its four neighbors, and evaluates the fitness of the action the neighbor is implementing using mental simulation. If its own action is fitter than that of the neighbor, it chooses another neighbor, until it has either observed all of its immediate neighbors, or found one with a fitter action. If no fitter action is found, the agent does nothing. Otherwise, the neighbor's action is copied to the input layer, learned, and implemented.

2.3.2.1 The Neural Network

The core of an agent is a neural network, as shown in figure 2.1. It is composed of six input nodes that represent concepts of body parts (LEFT ARM, RIGHT ARM, LEFT LEG, RIGHT LEG, HEAD, and HIPS), six matching output nodes, and six hidden nodes that represent more abstract concepts (LEFT, RIGHT, ARM, LEG, SYMMETRY and MOVEMENT). Input nodes and output nodes are connected to 'hidden' nodes of which they are instances (e.g. RIGHT ARM is connected to RIGHT.) Activation of any input node increases activation of the MOVEMENT hidden node. Opposite-direction activation of pairs of limb nodes (e.g. leftward motion of one arm and rightward motion of the other) activates the SYMMETRY node. The neural network learns ideas for actions. An idea is a pattern of activation across the output nodes consisting of six elements that instruct the placement of the six body parts.

The neural network starts with small random weights, and input patterns that represent ideas for actions are presented to the network. Ideas are learned by training the network for 50 iterations using the generalized learning rule [33]. Each time an input pattern is presented, the network's actual output is compared to the desired output. An error term is computed, which is used to modify the connectivity in the network such that its responses become more correct. Since the neural network is an autoassociator, training continues until the output is identical to the input, at which point the training stops. The value of using a neural network is that trends about what makes for a fit action can be detected using the symmetry and movement nodes (see below). The neural network can also be turned off to compare results to those obtained using a simple data structure that cannot detect trends, and thus invents ideas at random.

2.3.2.2 Knowledge-based Operators

Brains detect regularity and build schemas with which they adapt the mental equivalents of mutation and recombination to tailor actions to the situation at hand. Thus they generate novelty strategically, on the basis of past experience. Knowledge-based operators are a crude attempt to incorporate this into the model. Since a new idea for an action is not learned unless it is fitter than the currently implemented action, newly learned actions provide valuable information about what constitutes an effective

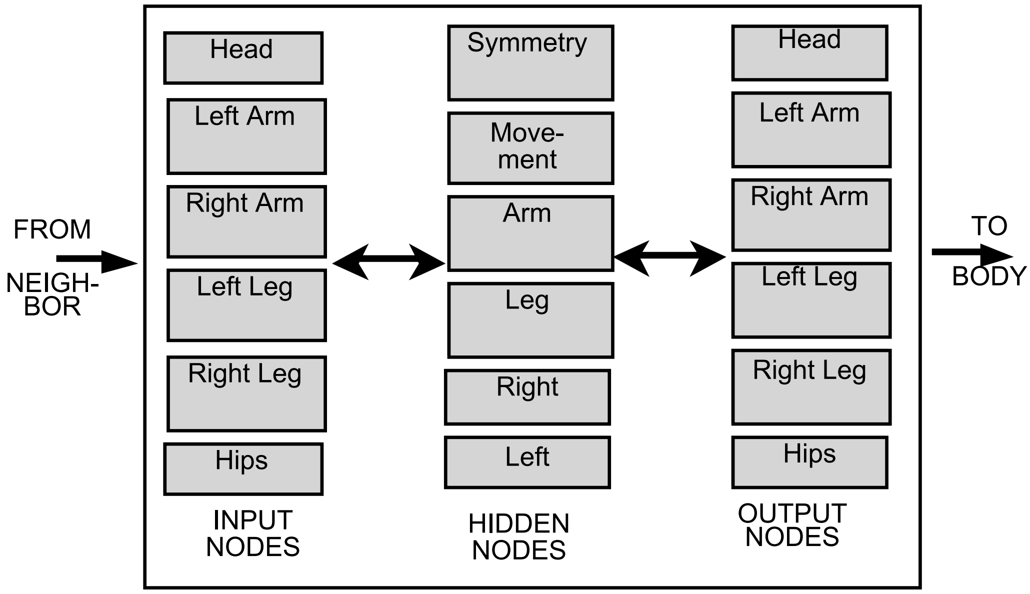


Figure 2.1: The neural network.

idea. This information is used by knowledge-based operators to probabilistically bias invention such that new ideas are generated strategically as opposed to randomly. Thus the idea is to translate knowledge acquired during evaluation of an action into educated guesses about what makes for a fit action. Two heuristics are used. The first is: if movement is generally beneficial, the probability increases that new actions involve movement of more body parts. Each body part starts out at a stationary rest position, and with an equal probability of changing to movement in one direction or the other. If the fitter action codes for more movement, increase the probability of movement of each body part. Do the opposite if the fitter action codes for less movement.

This heuristic is based on the assumption that movement in general (regardless of which particular body part is moving) can be beneficial or detrimental. This seems like a useful generalization since movement of any body part uses energy and increases the likelihood of being detected. It is implemented as follows:

$$P^{+}(i) = \begin{cases} max(1, P^{+}(i) + 0.1) & \text{for} \quad A_n \geq A_c \\ min(0, P^{+}(i) - 0.1) & \text{for} \quad A_n < A_c \end{cases} \tag{2.1}$$

$$P^{-}(i) = 1 - P^{+}(i) \tag{2.2}$$

with A_c = movement node activation for current action, A_n = movement node activation for new action, $P^{+}(i)$ = probability of increased movement at body part i and $P^{-}(i)$ = probability of decreased movement at body part i.

The second heuristic is: if fit actions tend to be symmetrical (e.g. left arm moves to the right and right arm moves to the left), the probability increases that new actions are symmetrical. This generalization is biologically sensible, since many useful actions (e.g. walking) entail movement of limbs in opposite directions, while others (e.g. pushing) entail movement of limbs in the same direction. This second heuristic is implemented in a manner analogous to that of the first. In summary, each action is associated with a measure of its effectiveness, and generalizations about what seems to work and what does not are translated into guidelines that specify the behavior of the algorithm.

2.3.2.3 The Body

If the fitness of an action is evaluated to be higher than that of any action learned thus far, it is copied from the output nodes of the neural network that represent concepts of body parts to a six digit array that contains representations of actual body parts, referred to as the body. Since it is useful to know how many agents are doing essentially the same thing, when node activations are translated into limb movement they are thresholded such that there are only three possibilities for each limb: stationary, left, or right. Six limbs with three possible positions each gives a total of 729 possible actions. Only the action that is currently implemented by an agent's body can be observed and imitated by other agents.

2.3.3 The Fitness Function

Agents evaluate the effectiveness of their actions according to how well they satisfy needs using a pre-defined equation referred to as a fitness function. The fitness of an action with respect to the need to attract mates is referred to as F1, and it is calculated as in [53]:

F1= $a_m + 2a_s + i$

with $i = 1$ if $a_h = 0$, or $i = 0$ otherwise.

Here, a_m is the activation of the movement hidden node, a_s the activation of the symmetry hidden node, and a_h the activation of the head node.

F1 rewards actions that make use of trends detected by the symmetry and movement hidden nodes and used by knowledge-based operators to bias the generation of new ideas. The application of F1 leads to actions inspired by realistic mating displays, and exhibits a cultural analog of epistasis. In biological epistasis, the fitness conferred by the allele at one gene depends on which allele is present at another gene. In this cognitive context, epistasis is present when the fitness contributed by movement of one limb depends on what other limbs are doing. In the simulations presented in the next chapter, F1 is used exclusively.

2.3.4 Incorporation of Cultural Phenomena

In addition to knowledge-based operators, discussed previously, agents incorporate the following phenomena, characteristic of cultural evolution, as parameters that can be turned off or on (in some cases to varying degrees):

- *Imitation.* Ideas for how to perform actions spread when agents copy neighbors' actions. This enables them to share effective, or 'fit', actions.
- *Invention.* This code enables agents to generate new actions by modifying their initial action or a previously invented or imitated action using knowledge-based operators (discussed previously).
- *Mental simulation.* Before implementing an idea as an action, agents can use the fitness function to assess how fit the action would be if it were implemented.

2.4 Implementation

EVOC is written in Java, an object oriented programming environment, using the Joone open source neural network library. The graphical user interface makes use of the open-source charting project, JFreeChart [67], enabling variables to be user defined at run time, and results to become visible as the computer program runs (figure 2.2).

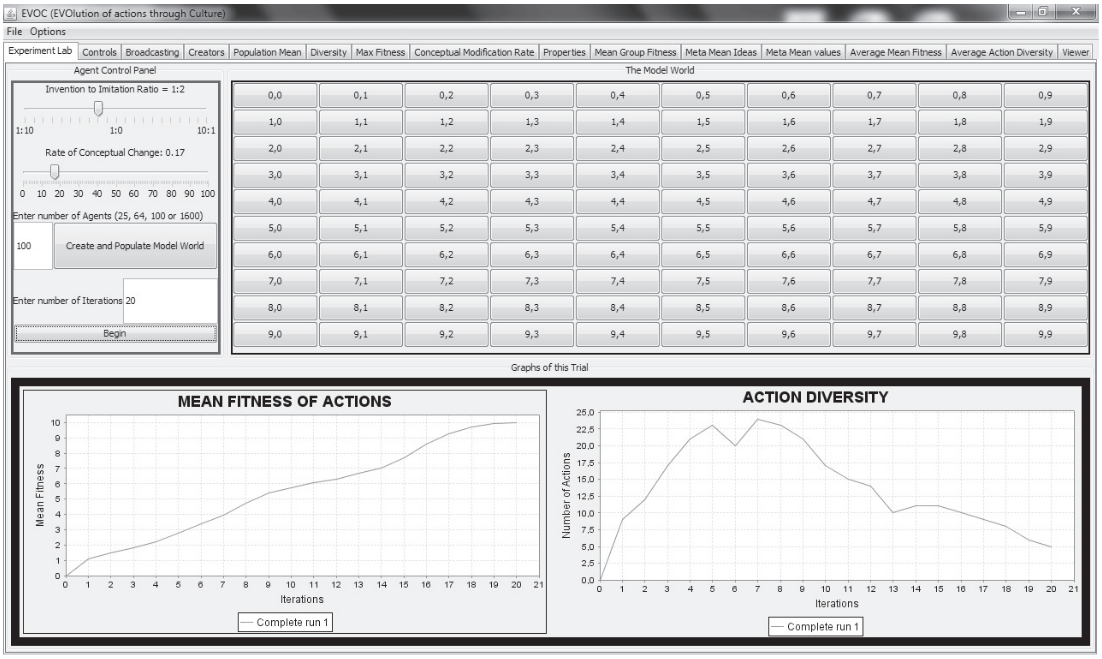


Figure 2.2: Output panel of GUI using F1.

2.5 A Typical Run

Fitness of actions starts out low because initially all agents start out in stationary rest positions. When an agent invents an action that has a higher fitness than doing nothing, this action will likely get imitated, so fitness increases. Fitness increases further as other ideas get invented, assessed, implemented as actions, and spread through imitation. The diversity of actions initially increases due to the proliferation of new ideas, and then decreases as agents hone in on the fittest actions.

3

CULTURAL EVOLUTION IN EVOC

In the experiments reported in this chapter[1], a distinction is made between two types of agents. Whereas one kind of agent, referred to as imitators, almost always obtains new ideas by imitating neighbors, the other type of agent, referred to as inventors or creators, almost always obtains new ideas by inventing them. There are two negative consequences of creativity in these simulations. The first is that an iteration spent inventing is an iteration not spent imitating. When invention does not lead to a more fit solution, this iteration could have spent better copying another inventor's idea, thereby increasing the probability of the persistence of this (potentially fitter) idea over time. The second is that creative change can break up co-adapted partial solutions:

[1]The results reported in chapter were originally published in the following articles (ordered by the section in which they appear): [107] Gabora, L. and Leijnen, S. (2009). The tradeoff between degree of creativity and number of creators in a computational model of society. In B. Cooper and V. Danos (Eds.) *Proceedings of Developments in Computational Models: Computational Models from Nature, July 11, 2009, Rhodes, Greece*; [106] Leijnen, S. and Gabora, L. (2009). How creative should creators be to optimize the evolution of ideas? A computational model. *EPTCS, 9*:108–119; [105] Leijnen, S. and Gabora, L. (2009). The artist loft effect in the clustering of creative types. In *Creativity and Cognition 2009*:389–390; [60] Gabora, L., Leijnen, S. and von Ghyczy, T. (2010). When Is Creativity Too Much of a Good Thing? A Computer Simulation. In *Proceedings of the 118th Annual Meeting of the American Psychological Association, August 12-15, 2010, San Diego, CA*; [61] Gabora, L., Leijnen, S. and von Ghyczy, T. (2013). Relationship between Creativity, Imitation, and Cultural Diversity. *International Journal of Software and Informatics, 7*(4):615–627, and [108] Leijnen, S. and Gabora, L. (2010). An agent-based simulation of the effectiveness of creative leadership. In *Proceedings of the Annual Meeting of the Cognitive Science Society, August 11-14, 2010, Portland, OR*:955–960.

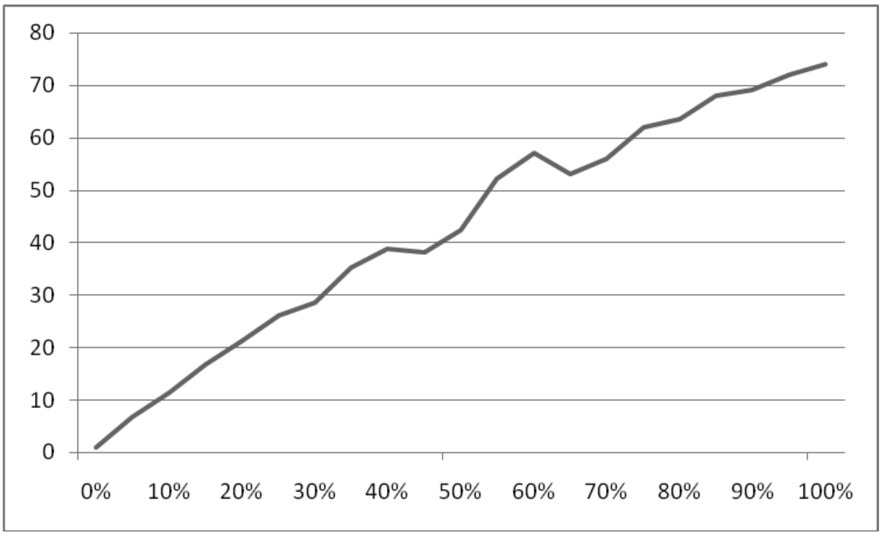


Figure 3.1: Action diversity after 50 iterations, for different percentages of creative agents in a population.

maximally fit actions are 'culturally epistatic', in the sense that what is optimal with respect to one part of the problem depends on what is done with respect to another part of the problem. Once both parts of the problem have been solved in a mutually beneficial way, too much creativity can cause these co-adapted solutions to break down. In this model of cultural evolution, epistasis is present when the fitness contributed by movement of one limb depends on what the other limbs are doing.

3.1 Proportion of Creators to Imitators

The first experiments show how much effect the ratio of creative individuals in a population has on the development of ideas. The world is not segmented, i.e. there are no barriers as in several other EVOC experiments reported elsewhere [53]. All experiments feature runs of 100 iterations and results displayed are averaged over 100 runs; on each run the creative agents are randomly dispersed. The creative agents provide for all the generation of new ideas, that is, imitators don't invent. Their role is to copy the successful innovations of other agents, and thereby serve as a 'memory' for preserving the fittest configurations. In this simulation, creators always innovate and never imitate; they have a 1:0 invention to imitation ratio. Conversely, imitators have a 0:1 invention to imitation ratio.

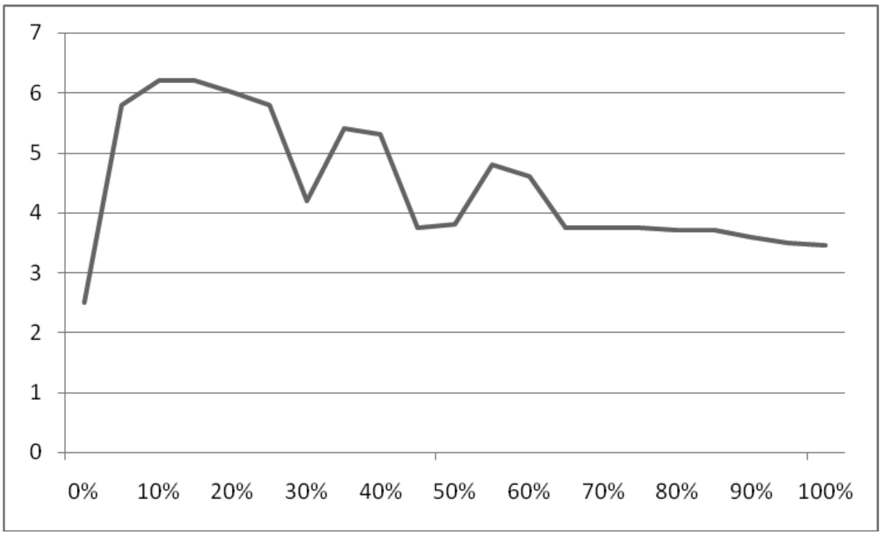


Figure 3.2: Mean fitness after 50 iterations, for different percentages of creative agents in a population.

Figure 3.1 illustrates that the action diversity (i.e. the number of different actions that exist in the world at a given time) in the artificial society is positively correlated with the percentage of creators. As the proportion of creators increases, a larger fraction of the search space is discovered. Figure 3.2 shows that although the number of configurations increases as a function of the proportion of creators, this does not necessarily have a beneficial effect on the mean fitness of the ideas in the society. If, for example, over twenty percent of the population exclusively engages in creative action, and the other eighty percent or less merely imitates, there are not enough imitators left to retain successful actions. In the long run this causes a disadvantageous effect on the mean fitness of ideas. The ideal population appears to consist of approximately 15% creators and 85% imitators.

3.1.1 Clustered and Unclustered Agents with Varying Degrees of Creativity

The world we have modeled supposes that agents can either innovate or copy actions, but not both. However, the real world is not so black-and-white; it contains individuals in all scales of gray, with some that tend to go their own way and do their own thing (but not always) and others that generally (but not always) follow trends. Using the

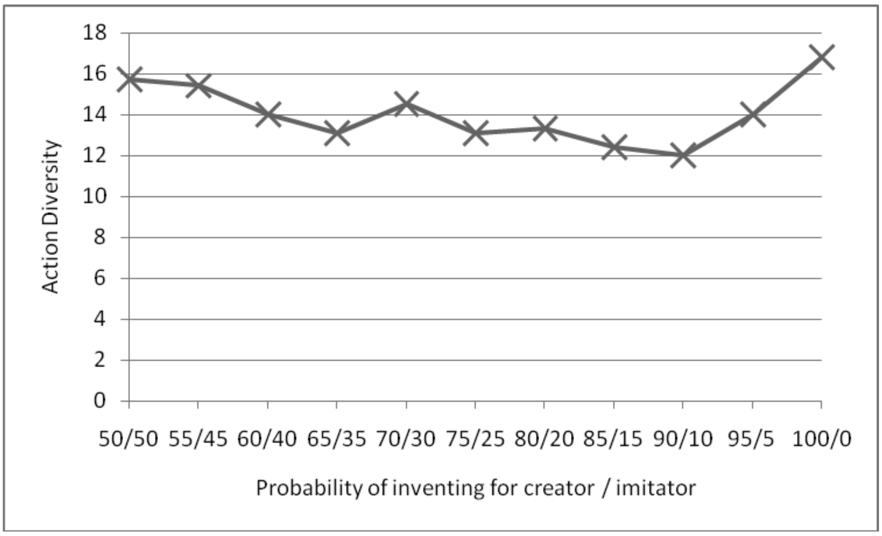


Figure 3.3: Action diversity after 50 iterations, for different probabilities of inventing versus imitating.

population distribution found in the previous experiment (fifteen percent creators), we varied the extent in which creators imitate and imitators innovate by changing their respective invention to imitation ratios. The results are depicted in figures 3.3 and 3.4.

From the results obtained after 50 iterations, the mean fitness and the action diversity show no clear trend of going up or down. The scores at 50/50 (both inventors and imitators have fifty percent probability of creating a new action) and 100/0 (inventors have a hundred percent probability of innovating; imitators have zero percent probability of innovating) are nearly the same. It appears that what matters most to the fitness of a population is not the number of creative individuals per se - but rather, the number of different ideas generated regardless of whether individuals exhibit individual differences as to their degree of creativity. As seen in figure 3.4, the diversity of ideas is also little affected by this variable. In other words, it does not appear to matter to the success of the society whether an idea comes from a regularly creative agent or an agent that usually imitates; these experimental results are discussed in section 3.1.2.

But what happens to the exchange of ideas when a group of creators engages in close interaction? It is not uncommon for creative individuals to work or live together within a group of likeminded, equally creative people. A synergy effect may emerge. The combined creativity of their works may benefit from such clustering. In order to

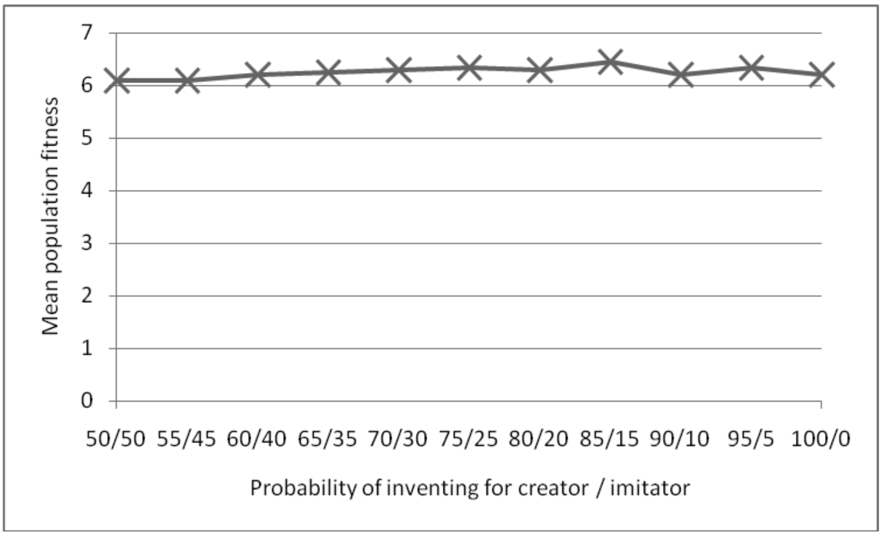


Figure 3.4: Mean fitness after 50 iterations, for different probabilities of inventing versus imitating.

simulate this, we predetermined the positions of agents, forcibly forming clusters of creators or putting them far apart.

As the parameters used (percentage of inventors in a population, innovation to imitation ratios) remain the same in between clustered and non-clustered experiments, the difference in mean population fitness can only be caused by the relative proximity of the creative agent (or agents). Figure 3.5 shows several world states after a number of iterations, with varying amounts of (clustered and unclustered) innovative individuals. The invention to imitation ratio used in these worlds was 1:0 for creative agents (and 0:1 for imitators). All novel actions are generated at one, two or three points in the grid, and spreads out over the population. This is particularly noticeable in the first column, where only one agent acts as a creator.

Using the rather rigid rule that creation means no imitation and imitation means no creation, the only agents that learn (the imitators) never put their knowledge into practice (they never invent some new action). For creative clusters to form, a certain interplay between the creators is required; they need to be able to learn from each other's inventions. We therefore gave creators a twenty percent probability of imitating, so that they had the chance to learn from their (possibly creative) neighbors. Figure 3.6 shows the results for a grid of 100 agents, of which four are creators, and figure 3.7 for a grid with 1600 agents, of which 10 are creators.

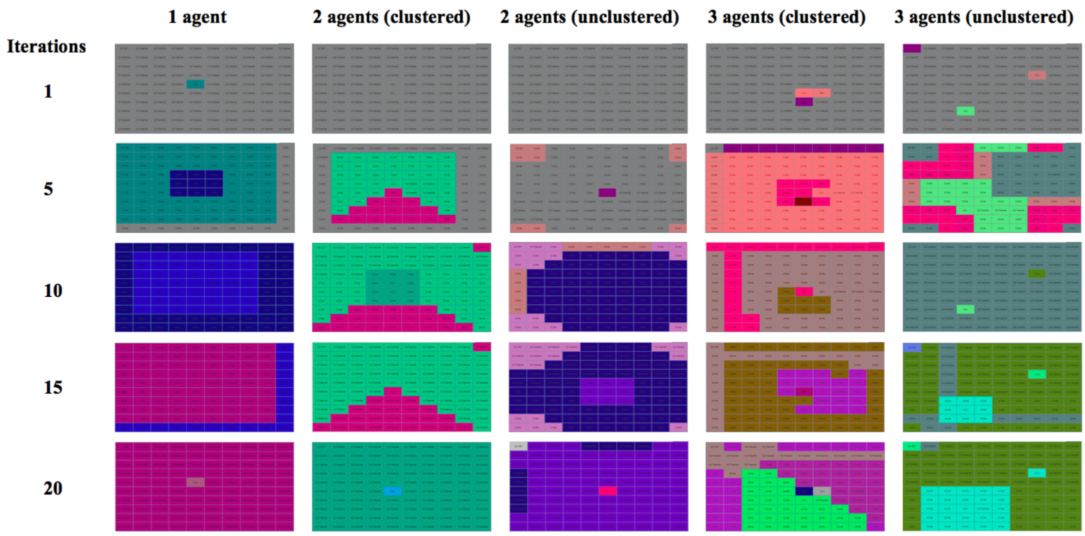


Figure 3.5: The development and transmission of actions in a world with 100 agents. Different actions are represented by differently colored cells. The snapshots of the worldgrid are ordered in time and in agent configuration. The columns indicate the state of the world after (from top to bottom) 1, 5, 10, 15 and 20 iterations. The rows show a configuration with (from left to right) one agent, two clustered agents, two unclustered agents, three clustered agents and three unclustered agents. Clustered groups of agents are placed next to each other somewhere in the (toroidal) world. Unclustered agents are placed in cells with maximum distance from each other.

Figure 3.6: The mean fitness of ideas developing over 30 iterations with a 100 agent grid, for clustered (solid line) and unclustered (dashed line) agents.

Figure 3.7: Mean population fitness developing over 30 iterations on a 1600 agent grid, for clustered (solid line) and unclustered (dashed line) agents.

These graphs show that clustering has a beneficial effect on the mean fitness of ideas in the artificial society, though with the very large society it appears to be detrimental in the short term. This may be because in such a large world too much clustering initially impedes the flow of ideas from creative to uncreative agents. The effects of clustering become more apparent in the larger grid world due to the relative distance between unclustered agents.

3.1.2 Discussion

It is known that creativity has serious drawbacks as well as benefits. The goal of the work reported here was to answer the question: when (if ever) does creativity become too much of a good thing? In the experiments reported here there is no possibility of newly invented ideas having unexpected negative side-effects or consequences. Nor is creativity associated with affective disorders such as depression. There are only two negative consequences of creativity in the simulations. The first is that an iteration spent inventing is an iteration not spent imitating. The second is that creative change can break up co-adapted partial solutions.

Previous work, using an agent-based modeling approach referred to as MAV, showed that the ideal ratio of imitating to inventing was approximately 2:1 [53]. That is, the fitness of ideas evolved most quickly when in 2/3 of iterations agents invented and in 1/3 of iterations agents imitated. In these experiments with MAV, all agents were equally capable of both inventing and imitating. In the experiments reported in this section we used EVOC to investigate a related but different question: what proportion of individuals should be ‘creative types’? The rationale is that it is known that in a society of interacting individuals capable of imitation, some members can capitalize on the benefits of creativity without incurring the drawbacks by merely imitating their creative peers. So if only some fraction of the population is a creator, and the rest imitators, what is the ideal ratio of creators to imitators? We found that when creators invent 100% of the time, and imitators imitate 100% of the time, we found that the ideal proportion is 15% creators versus 85% imitators.

Our subsequent experiment indicated that lowering the probability of inventing for creators while simultaneously increasing the probability of inventing for imitators (thus keeping the total amount of idea generation stable) does not have a significant effect on either the diversity or mean fitness of ideas in the population. This might appear to imply that it is not the creativity of individual agents that is important to the mean fitness of ideas in a group but rather the ‘overall creativity’ that is critical. We posited, however, the advantage of having specific individuals with more creative capacity than others may only become apparent when these highly creative individuals are not dispersed randomly but are close enough to one another to able to influence each other. Thus we hypothesized by clustering creative agents in EVOC such that they are more likely than not to be near other creators, creative ‘think-tanks’ might emerge with particularly good ideas, which spread for the benefit of all. This turned out to be the case. Creators and imitators alike adopted only the strongest ideas that came out of the think tanks (since weak ideas are quickly replaced by stronger ones). All this suggests that clustering may be a good way to maximize the mean fitness of ideas of a population. It is likely, however, that far more intricate configurations of agents lead to even better results. For instance, clustered individuals might influence each other to such a great extent that only one line of thought is explored (that of ‘the group’).

In the next sections, we will repeat the experiments undertaken to determine the ideal proportion of creators to imitators using creators that invent less than 100% of the time and imitators that do not imitate 100% of the time. This most extreme boundary condition was a good place to start, but not the most realistic. After all, in real societies all people, no matter how much they go their own way and do their own thing, imitate and adopt social conventions at least some of the time. Likewise no matter how uncreative people may be, they do adapt solutions to their particular circumstances and exhibit instances of 'mini-c creativity' on a daily basis. In the final section of this chapter, the 'broadcasting' capability of the program is used to investigate the question of what degree of creativity is associated with effective leadership.

3.2 Innovation Rate

The previous experiment shows how much effect the ratio of creative individuals in a population has on the development of ideas. The creative agents provide for all the generation of new ideas, that is, imitators don't invent. Their role is to copy the successful innovations of other agents, and thereby serve as a 'memory' for preserving the fittest configurations. In the simulations reported in this section, the rate at which creators innovate and imitate is varied. Imitators never innovate and always imitate.

Figure 3.8 shows that although the number of ideas increases as a function of proportion of creators, this does not necessarily have a beneficial effect on the mean fitness of the ideas in the society. If, for example, over twenty percent of the population exclusively engages in creative action, and the other eighty percent or less merely imitates, there are not enough imitators left to retain successful actions. In the long run this causes a disadvantageous effect on the mean fitness of ideas. The ideal population appears to consist of approximately fifteen percent creators and eighty-five percent imitators. When the society consists entirely of imitators, no new ideas are created and the fitness remains the same.

Figure 3.9 illustrates that the number of different actions in the artificial society is positively correlated with the percentage of creators. As the proportion of creators increases, a larger fraction of the search space is discovered.

Using EVOC to answer the question what proportion of individuals should be 'creative types' given a certain innovation rate, we found that when creators innovate more than 50% of the time, and imitators imitate 100% of the time, the ideal proportion is 15% creators versus 85% imitators. If the creativity ratio for creators is lower than 50%, the ideal percentage of creators increases toward 100%.

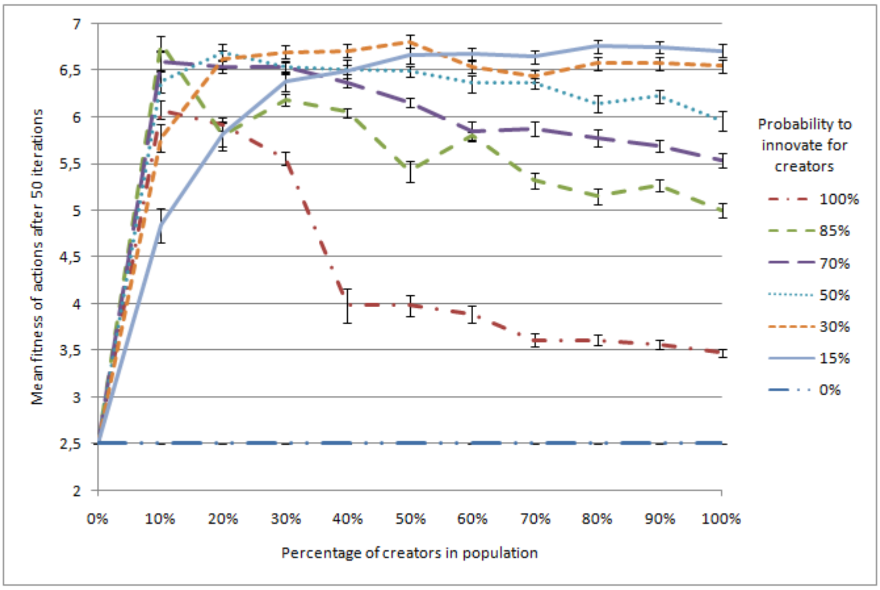


Figure 3.8: Effect of increasing percentage of creators in the population on mean fitness of ideas after 50 iterations, for different degrees to which creators are creative. Error bars indicate the standard deviation over 100 runs.

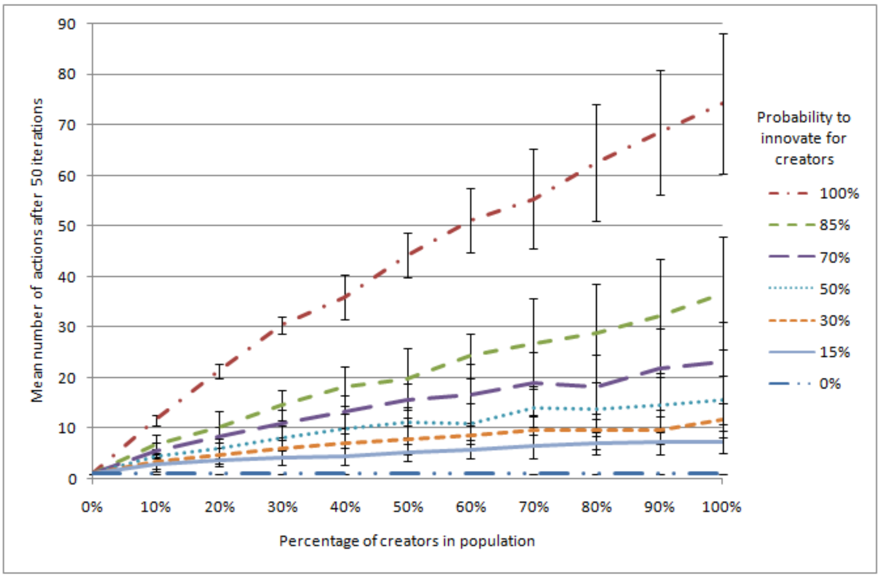


Figure 3.9: Effect of increasing percentage of creators in the population on diversity of ideas after 50 iterations, for different degrees to which creators are creative. Error bars indicate the standard deviation over 100 runs.

3.3 Time Series Analysis

The overall fitness of a population is a complex function of at least two critical parameters: the share of population engaged in innovation and the rate at which creative agents switch between innovation and imitation. The previous numerical simulations suggest that if the share of innovators is low, optimal results are reached when they dedicate themselves fully to innovation. However, as the number of innovators increases, the population is better off if innovators reduce their rate of novelty creation and spend time imitating other's ideas.

We investigated both the effect of varying how abundant creators are, and the effect of varying how creative they are. The frequency of creators - that is, the proportion of creators relative to imitators in the population - is referred to as C. The creativity of creators - that is, the probability that a creator invents a new action instead of imitating a neighbor - is referred to as p. It may help to think of the entire population as being divided into three subgroups at any given iteration:

- $C * p * 100$ agents are creative agents attempting to innovate;
- $C * (1 - p) * 100$ agents are creative but not attempting to innovate;
- $(1 - C) * 100$ agents always imitating.

It is also important to keep in mind that whereas the attributes creative and imitative are permanent, the subgroups of creative agents either innovating or imitating at any given time fluctuate stochastically. The process of innovation as driven by the neural network explained in the previous chapter. If the attempt to innovate is abortive, the agent retains it current configuration and fitness. If a creative agent does not attempt to innovate (with a probability of $1 - p$), it will behave as an imitator. The process of imitation is analogous to "lazy (non-greedy) search." The imitating agent will scan its four neighbors in random order, adopt the first configuration with a fitness greater than its own or, failing so, retain its current status.

3.3.1 Effects of Varying Creativity Rate and Proportion of Creators on Diversity of Actions

Figure 3.10 illustrates the average diversity, or number of different actions in the artificial society, over the course of a run as a function of C and p. Action diversity is

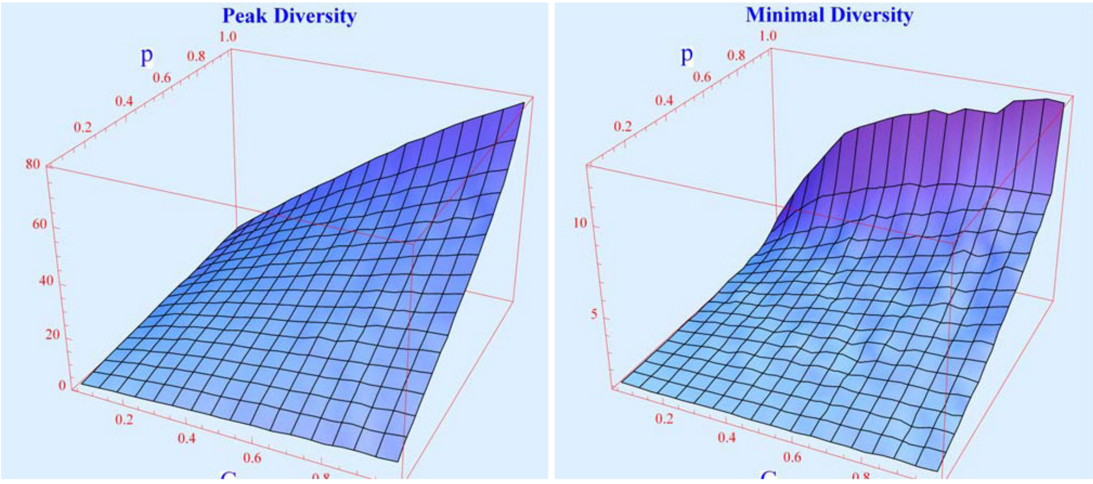


Figure 3.10: Average action diversity for different values of creator innovation probability (p) and creator-to-imitator ratio (C), showing maximum (left) and minimum (right) diversity.

positively correlated with both the percentage of creators, and their level of creativity. As C and p increase, a larger fraction of the search space is discovered. This generally holds true for both the maximum (peak) diversity as well as the minimal diversity during a run.

As the reader will note by comparing the results above with our findings on fitness (discussed below), high degrees of diversity are coupled with low overall fitness and vice versa. This appears to be at odds with the great adaptive benefit of diversity generally found in real-life evolutionary systems, be they of biological, cultural, economic or of some other nature. The divergence between model and reality in this respect is due to two interacting features of the model. First, whereas in real life, evolution unfolds in an opportunity space which may be infinite, agents in EVOC are limited to a fairly small number of potential configurations of which only a much smaller subset (8) are optimal. Hence, high fitness demands that diversity be forsaken, and the sooner the better. Second, imitation (in evolutionary systems of a social nature) and inheritance in biology are predominantly recombinatorial - that is, they need not reduce diversity and often function as the major source thereof. In contrast, imitation as presently implemented in EVOC is the imitator acquiring all features of the sole other agent selected for imitation - and thus effectively depletes diversity.

3.3.2 Effects of Varying Creativity Rate and Proportion of Creators on Fitness of Actions

For different values of C and p, the mean fitness of ideas over the course of a run is measured. Given such a set of fitness series of accumulating value over time, it is far from clear which series is the most desirable. The reason for this is, of course, that the series cannot be unambiguously ordered unless for each pair of series one strictly dominates the other. That is not the case for the results obtained with these simulations. The curves representing mean population fitness at different values of $\{C, p\}$ all increase monotonically but they often cross and re-cross as time progresses. What then is the optimal setting of $\{C, p\}$? This seemingly simple question will yield no clear answer unless we employ some assessment of the effect of time on the valuation of future benefits.

3.3.2.1 Time Series Discounting

The approach generally used to overcome this optimization problem is some form of discounting which associates a "present value" with any future benefit in such a manner that the present value of any given benefit diminishes as a function of the elapsed time until the benefit is realized. The standard approach in financial settings is exponential discounting. Given a series of benefits b_t, the Net Present Value (NPV) for N periods is defined as:

$$NPV(b) = \sum_{\mathrm{t=1}}^{\mathrm{N}} r^{t-1}\, b_t \quad \text{with} \quad 0 < r \leq 1 \tag{3.1}$$

The discount rate r is normally set as $r = \left(\frac{100+i}{100}\right)^{-1}$ where i is the interest rate (in percentage) for the unit period that an investor can obtain from a safe investment.

The basic idea of discounting may be used, albeit in a slightly altered form, to assess the mean population fitness at different values. Two such methods are presented in the next sections: Time-to-Threshold (TTT) discounting and Present Innovation Value (PIV) discounting.

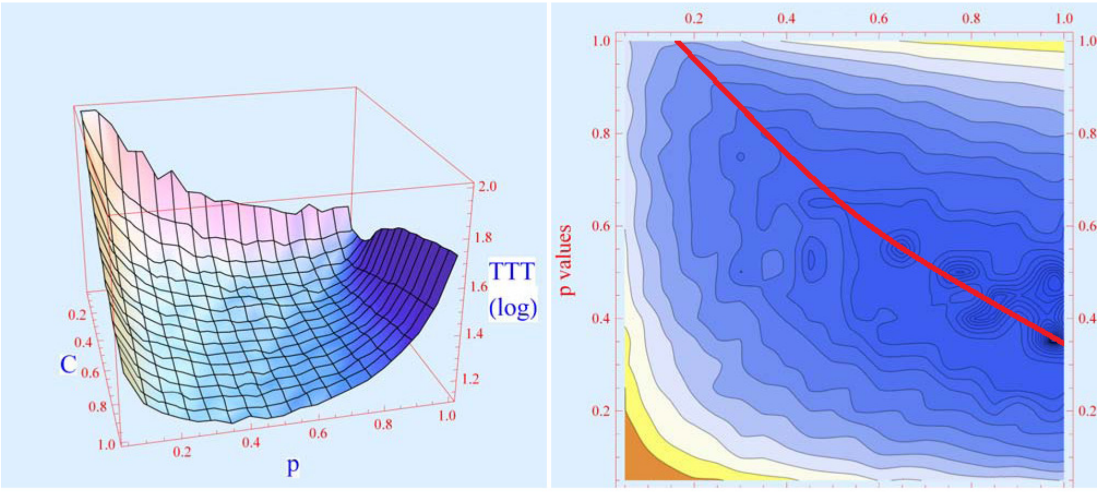


Figure 3.11: 3D graph (left) and contour plot (right) showing the $\log_{10}$ Time-to-Threshold (TTT) landscape of the average mean fitness for various values of C and p, with $\tau = 9$; the red line in the contour plot indicates the ridge.

3.3.2.2 Time-to-Threshold (TTT) Discounting

Since all fitness trajectories are monotonically increasing, those which reach a reasonably high threshold sooner should be valued higher by the agents. We therefore measure how many iterations (the time to threshold) it takes for fitness to reach τ. Figure 3.11 shows the $\log_{10}$ values for different ratio and probability values in a 3D graph and in a top-view contour plot for $\tau = 9$. Note that by definition, a low TTT value corresponds to high fitness.

Using the TTT method with this particular value for τ as a measure of optimal fitness not only allows for realistic averaging over time, it also clearly demonstrates the existence of a ridge in the landscape (indicated by a red line) which has been obtained by visually extrapolating over the minimal values. Given any given $C > 0.15$ or any given $p > 0.35$, the ridge determines which value of p or, respectively, C results in optimal population fitness. The global optimum is at approximately $\{C, p\} = \{1, 0.35\}$. For values of $C < 0.15$, fitness is optimal at $p = 1$. For values of $p < 0.35$, fitness is optimal at $C = 1$.

3.3.2.3 Present Innovation Value (PIV) Discounting

We note that innovative agents can flourish on their own whereas imitators can only do so in a society that attracts and retains innovative agents. In deciding whether or not to join and remain in the EVOC community, an innovative agent would naturally consider how well it could do on its own when fully engaged in experimentation (that is at $p = 1$) and compare that to what levels of fitness it is likely to acquire in a community of agents operating at some setting of $\{C, p\}$. Assuming that innovative agents fully subscribe to the tenets of strict economic rationality (which is not something they have inherited from their creators), they will desert if this comparison is negative and the society disintegrates. This suggests that in the framework of EVOC the fitness prospects of innovators working on their own is at least as important as the interest yield of treasury bonds in investment decisions. This logic leads to a simple modification of the standard discounting method for use in the context of EVOC.

Let $F_t^{C,p}$ be the mean population fitness at period t for the parameter setting $\{C, p\}$. Note that $F_t^{1,1}$ the fitness expectation of a solitary innovative agent because at this setting there is effectively no social interaction between agents. We can then define the Present Innovation Value (PIV), with N periods, for any fitness curve as:

$$PIV(F^{C,p}) = -N + \sum_{\mathrm{t=1}}^{\mathrm{N}} \frac{F_t^{C,p}}{F_t^{1,1}} \tag{3.2}$$

Naturally, we then have $PIV(F^{1,1}) = 0$, that is, the innovative agent is indifferent to either working alone or in a 'community' where everybody is working on its own and there is no social interaction in the form of imitation. Figure 3.12 shows the results for PIV discounting. Note that in these visualizations, the z-axis of PIV values has been reversed in order to obtain an unobstructed view into the surface. That is, lower values are better. Based on the logic developed above, innovators will not desert except when values of both C and p are very low. It is also clear that $C = p = 1$ is severely suboptimal, but one must also point out that the surface is flat over a large domain.

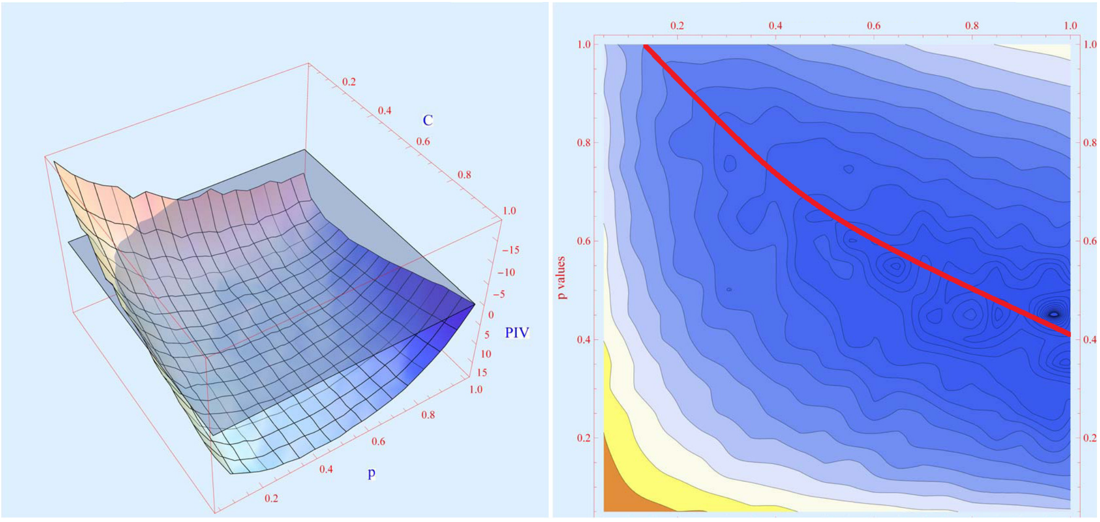


Figure 3.12: 3D graph (left) and contour plot (right) showing the Present Innovation Value (PIV) landscape of the average mean fitness for various values of C and p; the red line in the contour plot indicates the ridge.

3.3.3 Discussion

In this section, we report the results of an approach to this question that employed a detailed and sophisticated analysis using Time-to-Threshold (TTT) and Present Innovation Value (PIV). We found action diversity to be positively correlated with both the percentage of creators, C, and their level of creativity, p. When both $\log_{10}$ TTT values as well as PIV values for different ratio and probability settings were plotted in a 3D graph and a top-view contour plot, this revealed a clear ridge in the fitness landscape, indicating the optimal values of C and p.

The fundamental finding, in the simplest terms, is that imitation is neither just the greatest compliment nor - less charitably put - a form of free-riding, but an indispensable social mechanism that serves innovators and imitators alike. Without innovation, there is nothing to imitate, but innovation loses most of its potential to create welfare unless judiciously alloyed with imitation.

We note that the results obtained here reflect in part constraints in the current architecture of the model. EVOC in its current implementation does not accommodate combinatorial or selective imitation. An imitating agent copies all of the features of a single reference agent; it does not copy selectively from several agents, nor can it retain some of its own features. Consequently, imitation destroys diversity. At the same time imitation is essential to the rapid spread for superior configurations and steep increase of fitness.

3.4 Creative Leadership

It is widely assumed that effective leaders are creative [5, 7, 130, 142, 149]. However, we have seen that creativity can have drawbacks as well. The rationale is that in a group of interacting individuals, only a fraction of them need to be creative for the benefits of creativity to be felt throughout the group. The rest can reap the benefits of the creator's ideas by simply copying, using, or admiring them. After all, few of us know how to build a computer, or write a symphony or novel, but they are nonetheless ours to use and enjoy. Numerical simulations showed that if the proportion of creators is low, the mean fitness of ideas in the artificial society is highest when creators dedicate themselves fully to invention. However, as the proportion of creators increases, for optimal results, creators should spend more time imitating. Creative agents amounted to 'puncture points' in the fabric of society that interfered with the dissemination of proven effective ideas.

In this section, we focus exclusively on the extent to which creativity is desirable in a leader, where leadership is equated with having substantial influence over others. Previous results indicated that the presence of a leader accelerates convergence on optimal ideas, but does so at the cost of consistently reducing the diversity of ideas [57, 58]. In these previous simulations, the leader was no more nor less creative than the rest of the agents, referred to here as followers. The goal of the work reported here was to investigate how creative versus uncreative leadership affects the group as a whole.

The creative leadership experiments described below all make use of EVOC's broadcasting function. Broadcasting enables the action implemented by a leader to be visible throughout the artificial society. While previous experiments investigated the impact of varying the number of leaders on the fitness and diversity of ideas, in the experiments reported here, all simulated societies consist of one leader and ninety-nine followers. This leader is chosen once randomly and broadcasts throughout the entire run. In previous experiments, agents choose the fittest action among those of their neighbors

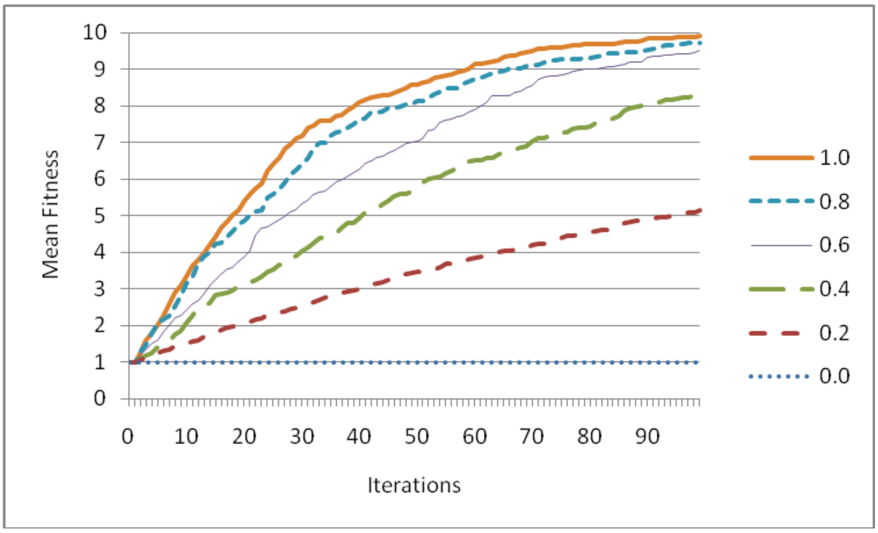


Figure 3.13: Mean fitness of actions with leaders of varying invention-to-imitation ratios, and followers that only imitate, i.e. that never invent (e.g. $i = 0.0$). The legend indicates the values of i_{leader}.

when imitating. Here, the leader acts as a neighbor to all other agents, as its actions are also taken into account during imitation.

3.4.1 Effect of Varying Inventiveness of Leaders and Followers on Fitness of Ideas

The first experiment investigated the effect of varying the ratio of iterations spent inventing versus imitating, or invention-to-imitation ratio, abbreviated i, of both the leader and the followers, on the fitness of ideas produced by the artificial society. The inventiveness of the leader, abbreviated i_{leader} was systematically varied from 0.0 to 1.0. When i_{leader} was 1.0, the leader invented a new action every iteration. When i_{leader} was 0.0, the leader never invented new actions; it only imitated its neighbors' actions. It was still the leader because its actions were visible to, and could be imitated by, all other agents in the society, not just its immediate neighbors, as was the case for followers.

In the first set of runs, followers imitated each iteration and never invented, i.e., $i_{follower} = 0.0$. As shown in figure 3.13, with uncreative followers, the degree of creativity of the leader mattered a lot; the mean fitness of ideas across the artificial society was positively correlated with the creativity of the leader.

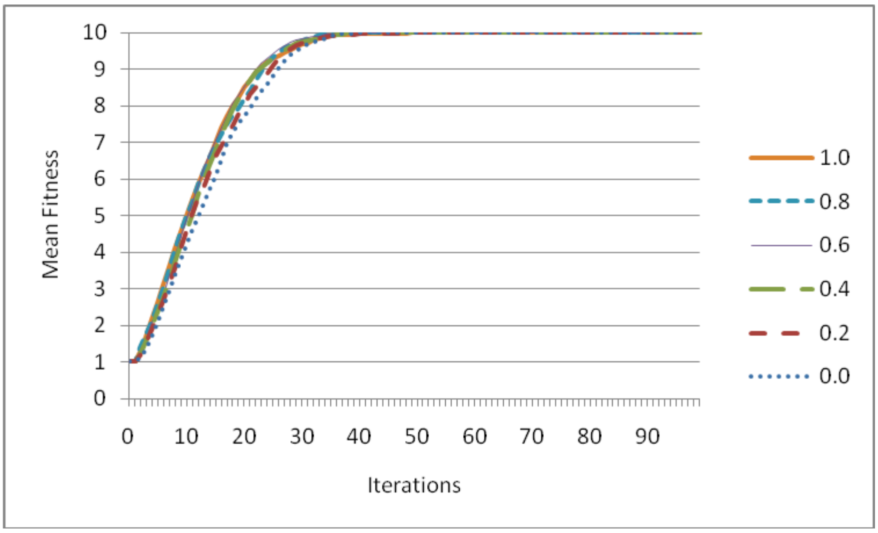


Figure 3.14: Mean fitness of actions with leaders of varying invention-to-imitation ratios, and followers that invent as well as imitate ($i = 0.05$). The legend indicates the values of i_{leader}.

In the second set of runs, shown in figure 3.14, followers were able to invent. More specifically, $i_{follower} = 0.05$; thus in each iteration, each of the 99 followers had a 5% chance of inventing. Comparing figures 3.13 and 3.14 it is clear that while the degree of creativity of the leader had a large impact when followers are uncreative, it had almost no impact when followers were themselves creative. With creative followers, the mean fitness of ideas generated by the society increased over the duration of a run at more or less the same pace no matter how creative the leader was.

3.4.2 Effect of Varying Inventiveness of Leaders and Followers on Diversity of Ideas

The second part of this experiment involved investigating the effect of varying the invention-to-imitation ratio i, of both the leader and the followers on the diversity of ideas produced by the artificial society. As in the previous experiment, i_{leader} was systematically varied from 0.0 to 1.0. The result obtained with $i_{follower} = 0.0$ is shown in figure 3.15.

In the short run, creative leadership was associated with increased diversity of actions. However, in the long run, no matter how creative the leader, all agents converged on the same action, despite that there were seven other equally optimal actions they

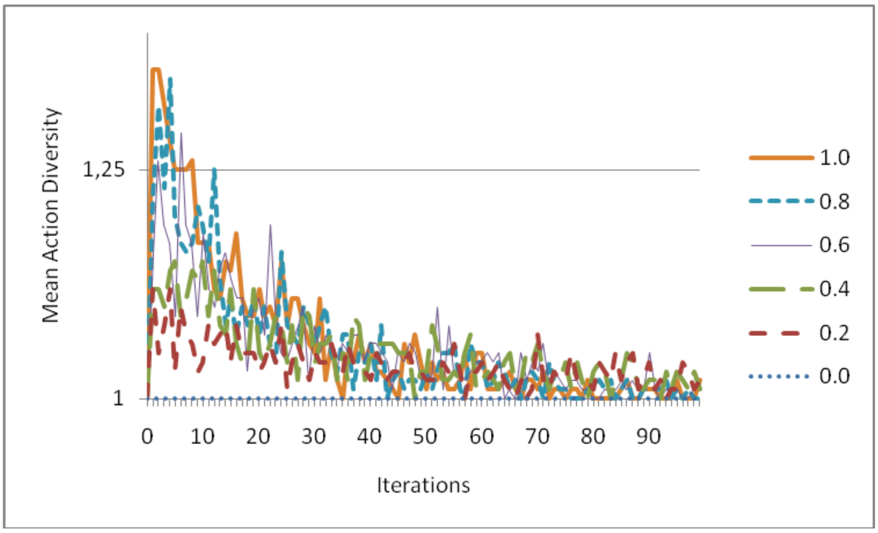


Figure 3.15: Diversity of actions in the artificial society with leaders of varying invention-to-imitation ratios, and followers that only imitate ($i = 0.0$). The legend indicates the values of i_{leader}.

could have converged upon. Results with higher values of $i_{follower}$ (not shown) were qualitatively similar. Action diversity was initially substantially higher, but it still always eventually converged to 1.

3.4.3 Effect of Varying Leaders' Rate of Conceptual Change

There are two ways an agent's creativity can be manipulated in EVOC. The first way involves changing i, the invention-to-imitation ratio, as in the first set of experiments. It is possible to vary not just how frequently an agent invents, but how creative its newly invented ideas are. This second measure, referred to as the rate of conceptual change, abbreviated c, is implemented as follows. Invention occurs by taking the current action, and modifying it. When c is low, the newly invented action varies little from the previous action upon which it was based. When c is high, the newly invented idea varies dramatically from the previous idea upon which it was based.

As mentioned previously, the default value of c, the probability of change to any body part during invention, is 1/6 for any agent that invents, whether it is a leader or a follower. Previous experiments revealed this to be the rate that optimizes the rate of increase in mean fitness of actions [53]. Since ideas are ideas for actions, and since actions involve at most six body parts, on average, each newly invented action involves

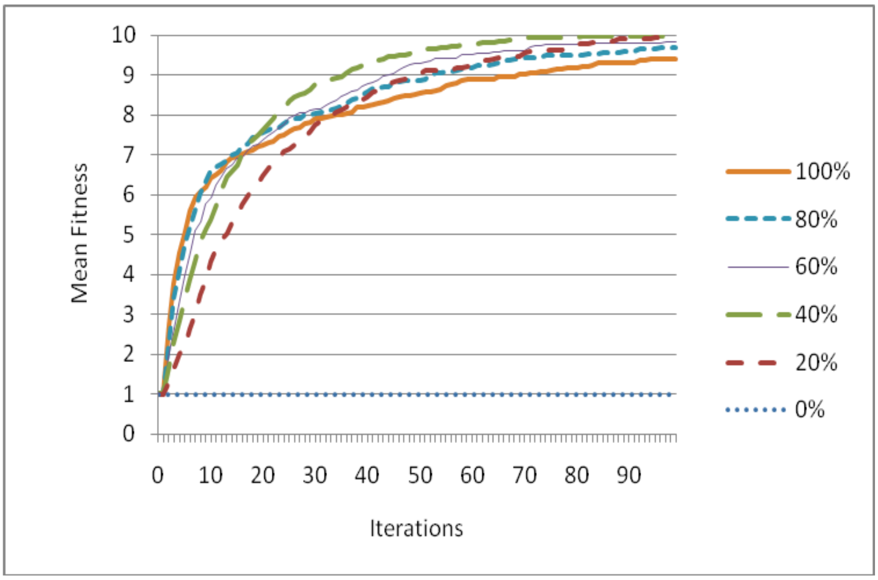


Figure 3.16: Mean fitness of actions in the artificial society with leaders of varying rates of conceptual change, and followers that only imitate. The legend indicates the values of c_{leader}.

a change to the motion of one body part. Thus $c = 1/6$ means that each body part changes what it is doing with a 1/6 probability, or 17% of the time. In this second set of experiments, shown in figure 3.16, c_{leader} was systematically varied from 0% to 100%. Since the followers only imitated, $c_{follower} = 0$. Because that means there are no new actions for the leader to imitate, i_{leader} was set to 1.0.

Unlike the experiments reported previously, the optimal degree of creative leadership with respect to this second measure of creativity depends on what phase of the creative process the society was at. Early on in a run, a form of leadership that entails the highest possible rate of conceptual change (100%) was most beneficial. However, as the run progressed a transition occurred, after which point a much lower rate of conceptual change (approximately 40%) was most beneficial.

3.4.4 Discussion

The experiments reported in this section investigated the impact of creative versus uncreative leadership on the mean fitness and diversity of ideas for actions in an agent-based artificial society. The first experiment looked at the effect of varying the invention-to-imitation ratio of both leader and followers. The mean fitness of actions was positively correlated with the creativity of the leader, but only when the followers were uncreative. The more creative the followers, the greater the extent to which the beneficial effect of creative leadership was washed out. Although one must be cautious about extrapolating from a simple simulation such as this to the real world, the result suggests that creativity may be a relatively unimportant quality for a manager of a creative team, but an important quality for a manager of an uncreative team.

In the first experiment, we also investigated the effect of varying the invention-to-imitation ratio of both leader and followers on the diversity or number of different actions implemented by agents in the artificial society. Previous results with EVOC had suggested that the beneficial effect of leadership on mean fitness of ideas is tempered by decreased diversity of ideas, and this echoed previous simulation findings that leadership can have adverse effects when agents can communicate [66]. We wanted to know whether the decreased diversity associated with the presence of a leader was still observed when leaders are highly creative or highly uncreative compared to followers. We found that while in the early stages of a run, creative leadership (as well as the degree of creativity of followers) was associated with higher diversity, eventually all agents converged on what the leader was doing no matter how creative the leader (or how creative the followers). This suggests that in the long run leadership diminishes cultural diversity regardless of how creative the leader is. It is worth noting, however, that in this artificial world, unlike the real world, agents had only one task to accomplish; further research is required to investigate whether these results hold true when the fitness function varies over time.

The second set of experiments investigated the effect of not how often the leader invents, but how original any particular invention is, referred to as the rate of conceptual change. We found that early on in the creative process, when the fitness of the ideas that are getting generated was still relatively low, it was best if the leader was very creative (high rate of conceptual change). However, later in the creative process, once relatively fit ideas were being generated, a less creative leader was better (low rate of conceptual change). This result may reflect that the fitness function used here exhibits 'cultural epistasis'. Initially, the higher the rate of conceptual change, the more quickly fitter actions are found. However, once relatively fit actions have been found, a high rate of conceptual change breaks up co-adapted epistatically linked elements and thus interferes with convergence toward optimal actions. We suspect that many real-world problem solving situations involve this kind of epistasis. Thus, although once again one must be cautious about extrapolating from the results of simple simulations such as this to the real world, our results suggest that a new startup company benefits most from highly creative leadership, while a more established company, or one that has stabilized on an established product line, benefits most from a more conservative form of leadership.

PART II

CREATIVITY AND CONSTRAINT

4

CREATING SYMBOLS

This chapter[1] explores the difference between indexical and symbolic interpretation using a neural network simulation of a series of language training experiments with chimpanzees. The results of this simulation lead to a discussion about the systemic requirements for crossing the symbolic threshold and how the primacy of icons applies to computational models. Whereas modeling tasks that require iconic interpretation (such as pattern recognition and classification) are well-understood in the computational domain, modeling indexical and particularly symbolic interpretation seems to require a hierarchical system dynamics that doesn't readily fit within the existing computational and neural network paradigms.

4.1 The Meanings of Symbol

In a study aiming to test the linguistic abilities of chimpanzees, several experiments are devised and conducted to demonstrate how different learning strategies produce different uses of language [137]. The study shows how their learning curves can be understood from the way these chimps acquire language, allowing for a behavioral operationalization of language acquisition. The results are embedded within a larger semiotic theory of symbolic interpretation, distinguishing between three types of signs

[1]This chapter appeared as [104] Leijnen, S. (2012). Emerging symbols. In T. Schilhab, F. Stjernfeld and T. Deacon (Eds.) *The Symbolic Species Evolved*. Springer Biosemiotics Series, 6:253–262.

(icons, indices and symbols) that describe how an object can be related to a referent by an interpreter [20, 23, 83].

Several other language training studies [62, 128, 136] show that apes can acquire large vocabularies. The subject has to point to one or more lexigrams on a board in order to express its thoughts or desires. Researchers stimulate the apes to use the correct lexigrams and apply appropriate grammar rules. However, even though their sentence construction capability can be trained to be more or less flawless, their learning strategy appears to differ from the way humans would approach such a problem. Although they appear to use lexigrams as representations of the objects they stand for (like humans do) their pointing behavior is a trained response to the presented stimulus.

The often implicit assumption that these apes use lexigrams as representations for something else is not to be easily overlooked. For us to talk about apes using language and having a vocabulary, evidence is required that - indeed - these apes use linguistic skills to solve a problem, instead of associative skills to merely discover a correlation between stimuli and responses leading to a reward. The difference between these two skills is subtle but crucial, especially considering the principal reason for doing ape language studies is finding out if they are actually capable of learning a language.

So how are we to make this distinction clear? We find two contrasting definitions of symbols in which the difference is expressed [36]:

S1 a symbol is one of a conventional set of tokens manipulated with respect to certain of its physical characteristics by a set of substitution, elimination, and combination rules, and which is arbitrarily correlated with some referent;

S2 a symbol is one of a conventional set of tokens that marks a node in a complex web of interdependent referential relationships and specific reference is not obviously discernible from its token features. Its reference is often obscure, abstract, multifaceted, and cryptic, and tends to require considerable experience or training to interpret.

The chimpanzees in the Savage-Rumbaugh & Rumbaugh study [137] are subjected to a training program that causes the disparity between these two kinds of symbols to become salient, demonstrated by a significant difference in performance results. In one experiment, the chimps learn to distinguish lexigrams for four objects (banana, orange, coke and milk) and two verbs (give and pour). The chimpanzees are required to use the correct verb with each noun by arranging them in a sentence. Producing accurate sentences like give orange or pour milk is rewarded; producing incorrect compounds like pour banana or coke milk is discouraged.

Once the chimps have learned to associate pairs correctly, a follow-up experiment shows that their symbol use is, in fact, non-symbolic. As the researchers introduce new edibles and liquids to the experiment, the amount of trials needed to learn to embed these words into sentences grows. Instead of using the web of relations to which the lexigrams refer - the chimps know that edibles are given and liquids are poured, but they don't apply this knowledge to the construction of lexigram sentences – they memorize each verb-noun correlation as a rule. The chimps use lexigrams as

> "[...] a set of events which come to precede the receipt of a desired action or object. [...] errorless trials, though given in a fashion which closely approximates that of the final choice, do not lead to symbolic learning even in simple tasks such as food names [137]."

The apes have learned to use symbols as defined by S1, but not according to the more strict definition S2. The relations between the lexigrams are arbitrary, as the chimps fail to notice the analogy with the relations between objects and actions. S1 is a rather shallow, computational definition of symbols that doesn't capture the way humans use symbols as expressed in S2. Hence, phrased in semiotic terms, the chimpanzees have learned to use lexigrams as indices. An index pairs two things together based on their co-appearance, like a thermometer (number and temperature) or a windsock (position and wind direction). In this case, a noun lexigram is paired with a verb lexigram.

For the ape subjects to use the lexigrams as symbols (according to S2) a reference is required to the network of relations for which the lexigrams stand. Evoking such a reference is exactly the goal of the next experiment in the chimp language training program. It is set up in almost the same way as in the previous ones, but this time the apes' attention is drawn toward the food and drink dispensers by increasing their saliency with visual and auditory signals. The apes now notice the dispensers opening, also when they're empty. This causes some of the apes to pair their understanding of objects and actions with their understanding of lexigrams, and transfer knowledge between these networks. Instead of memorizing each and every lexigram combination as an index, these chimps have created a symbolic link, which offers them a more efficient way of storing information in the long run.

4.2 Simulated Learning

The chimp language training research supports the claim that symbolism is not intrinsic to a word, lexigram or object, but is dependent on the interpretation itself. Interpreters can be iconic, indexical and symbolic. Some of the apes were capable of all three of these skills, while others could only reach the indexical level. In order to explain this gap, it would be insightful to see inside a chimp's head, study how signals travel between neurons and how eventually a lexigram sentence comes about. In a meticulous study of the chimp's interpretation process, the differences that cause the symbolic shift could be unveiled. Of course, the sheer complexity and size of the brain would result in far too many parameters for us to make sense of. As an alternative, computer simulated models of smaller, less complex brains can be used in order to discover the systemic requirements for symbolic interpretation.

For our experiments, we will use an artificial neural network: a three-layer perceptron [13] with full connectivity (figure 4.1). The nodes in the hidden and output layer are implemented with a step activation threshold function:

$$\alpha_i = \sum_{\mathrm{i=1}}^{\mathrm{n}} w_i \, x_i \ ,$$

$$y_j = \begin{cases} 1 & \text{for} \quad \alpha_i \geq \theta \\ 0 & \text{for} \quad \alpha_i < \theta \end{cases} ,$$

where α_i is the total activation of unit i, y_j the base value for output connection j, x_i the base value for input connection i, w_i the weight of input connection i, n the number of input connections and θ the threshold parameter (0.85).

By varying the connection weights between neurons, different network architectures are generated, each with a potentially different behavior (i.e. returning a specific output in response to a certain input). After a set of random weight configurations has

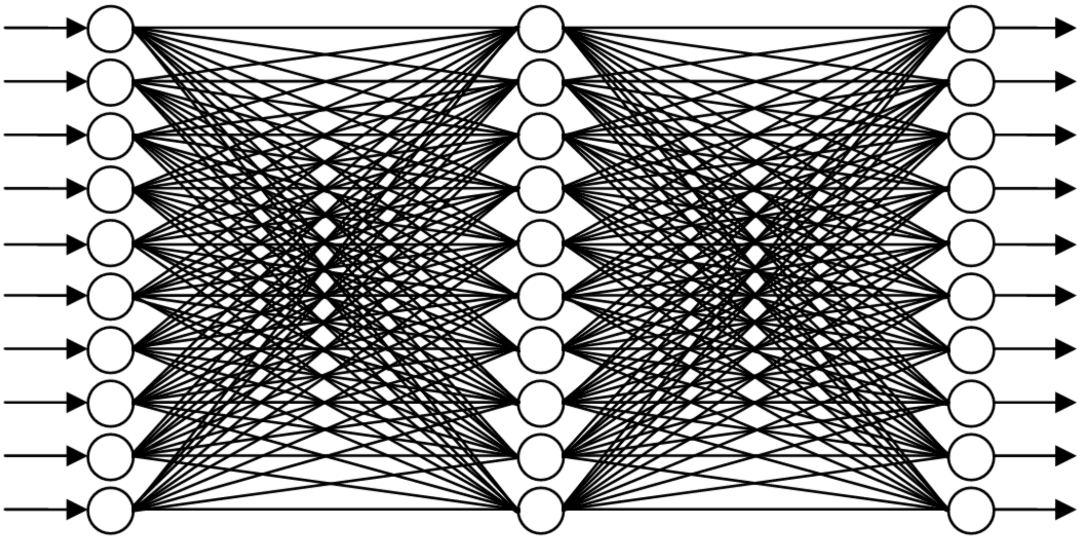


Figure 4.1: Three-layer perceptron will full-connectivity

been selected, each of their input layers is activated with trial data and propagated as an activation wave through the network. Weight configurations are stored in a binary array. A score is awarded to each network based on the percentage of desired output values in a series of training sessions. The highest scoring networks are then subjected to crossover - randomly combining the binary arrays of two successfull networks - and mutation - randomly flipping some bits of the binary arrays formed by crossover - to form a new generation of network configurations, and so on. Due to the similarity with biological evolution and the storage of information in gene-like data arrays, this method is formally known as a genetic algorithm [82]. The parameters of this particular GA are given in table 4.1.

Parameter	Value
children per generation	50
elites per generation	10
maximum generations	30000
learning runs	100
$P_{mutation}$ per bit	0.01

Table 4.1: Genetic algorithm settings.

4.3 Experiments

Using the computational tools described above, the difference between indexical and symbolic interpretation is shown in a series of experiments. The two types of chimps (symbolic and non-symbolic) of the original language training research are modeled as neural networks. Objects, actions and lexigrams are replaced by binary strings of input and output data. The genetic algorithm acts as a training program, forwarding input data into the networks and evaluating the results.

For the indexical learning model, the objects, actions and lexigrams are coded according to the method displayed in table 4.2. There are a couple of things that should be noted about this encoding. First, it disregards iconic interpretation processes by translating multifaceted entities into easily discernable icons. The chimpanzees are required to make distinctions between bananas, yellow lexigrams, cans of coke and acts of pouring, but the neural network simply uses a ten bit binary string as input and output of the indexical process. This ensures that the neural network learns to create indexical associations, instead of a mixture of icons and indices: marginalizing the role of iconic interpretation isolates the indexical interpretation process which facilitates the study of its features. Also, in order to allow for a fair comparison with the symbolic network (the number of activated input neurons may influence the learning rate, and therefore needs to remain constant in each experiment) a bias unit is added to the input vector.

Network input	Binary string	Correct output	Binary string
banana + bias	1000000001	banana lexigram + give lexigram	1000000010
coke + bias	0100000001	coke lexigram + pour lexigram	0100000001
orange + bias	0010000001	orange lexigram + give lexigram	0010000010

Table 4.2: Binary encoding examples for the indexical experiment.

The neural network is trained by the genetic algorithm to output the correct binary string, given a certain input string. For the input string, the leading eight bits indicate the presence of a particular edible or liquid, the ninth bit is always zero and the tenth bit is always one. The output string uses the leading eight bits to signify the use of a food or drink lexigram. The trailing two bits denote the use of an action lexigram. Once the first pairing has been learned (i.e. banana with give banana), a second pair

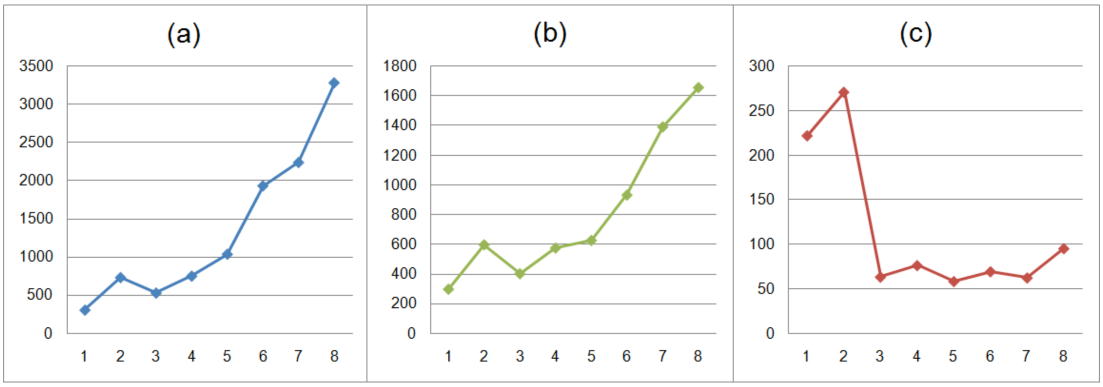


Figure 4.2: Learning curves for the indexical task (a), the symbolic task (b) and the domain task (c). The y-axis indicates the number of generations it takes for each additional object (x-axis) to be learned.

is added to the dataset. The learning continues with the same network and a training set of two possible input strings. This process is repeated until all eight objects have been associated with correct output sentences. The time it takes the network to learn each additional object is displayed in figure 4.2a. Notice that, for each task, learning the second object takes a considerable amount of time: after the network is trained to recognize one object exclusively, it takes time to learn that multiple objects need to be recognized. Adding a third requires less time learning, as the network is already trained to recognize multiple objects.

The chimps that learn to manipulate lexigrams as symbols are induced to adopt a new learning strategy by the food and drink dispensers. These dispensers make them reconsider the relation between the lexigram buttons and obtaining a reward. They notice a systemic similarity between the system of lexigrams and the system of objects and actions [34] and use their existing knowledge of the object domain to produce correct lexigram sentences.

For the symbolic learning model, we use the same approach as for the indexical simulation, with the exception of the domain knowledge being available in the input string. In other words, the subject already knows that a banana is given (not poured) and takes this knowledge into account when it constructs a sentence. The additional information helps to predict the correct outcome, as actions and action lexigrams are correlated. The training data is shown in table 4.3, the resulting learning curve in figure 4.2b.

Network input	Binary string	Correct output	Binary string
banana + give	1000000010	banana lexigram + give lexigram	1000000010
coke + pour	0100000001	coke lexigram + pour lexigram	0100000001
orange + give	0010000010	orange lexigram + give lexigram	0010000010

Table 4.3: Binary encoding examples for the symbolic experiment.

A comparison between the learning curves of the indexical and symbolic models is somewhat biased. Just as the chimpanzees were at some point required to learn that bananas are given and milk is poured, so should the symbolic network, one could argue. The goal of these experiments is to test the difference between indexical and symbolic learning; to exclude learning the domain knowledge would be a bias. Therefore, we carried out a third experiment. A neural network learns to associate objects with corresponding actions, using the same method as in the previous experiments. Table 4.4 contains the training data, the resulting learning curve is displayed in figure 4.2c.

Network input	Binary string	Correct output	Binary string
banana + bias	1000000001	give + bias	1000000001
coke + bias	0100000001	pour + bias	0100000001
orange + bias	0010000001	give + bias	0010000001

Table 4.4: Binary encoding examples for the domain experiment.

4.4 Conclusions

A neural network model is used to simulate two different learning strategies in a series of three experiments. A genetic algorithm operates on a population of networks to train them in producing the desired output string. To generate a training dataset with input and output patterns, eight objects, two actions and ten lexigrams that were also used in the chimpanzee trainings tasks are encoded into binary patterns. For each of the experiments, this results in a learning curve, showing the average number of generations needed by the genetic algorithm to find a working network configuration when a new object is inserted into the training dataset. The first experiment (indexical task) simulates how much learning time is required to map objects to lexigram sentences. In

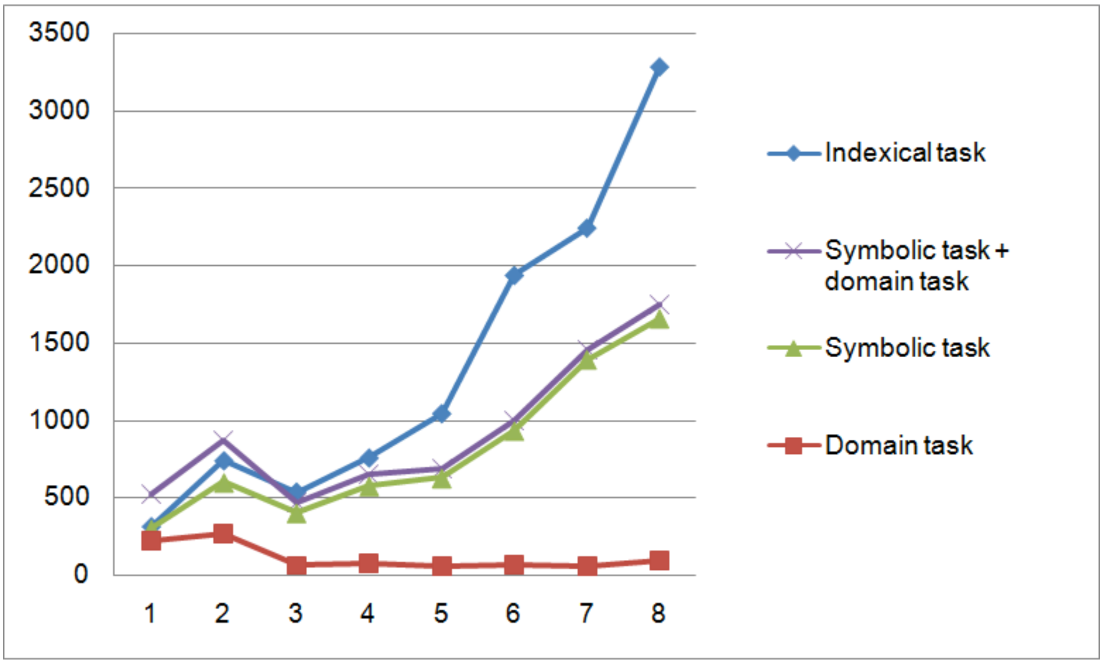


Figure 4.3: Learning curves for all three tasks compared. The y-axis indicates the number of generations it takes for each additional object (x-axis) to be learned.

the second experiment (symbolic task) both the object and the action are part of the input. Finally, a third experiment (domain task) is added to avoid a possible bias. In comparing the indexical and symbolic task, the learning time required for the domain knowledge task is added to the learning time for the symbolic task. This gives four learning curves, as shown in figure 4.3.

Several conclusions can be drawn from these curves. The domain knowledge task takes considerably less time than the other tasks, which can be attributed to the required output containing only one variable (either give or pour) instead of two. Also, there is an overall decrease in learning time after the third object is added. Once the two possible output patterns have been learned, the network has created a tendency to produce the right kinds of output patterns in the future. This holds for the indexical and symbolic tasks as well as for the domain task; however, due to the steep learning curves of the former two, this effect is not as significant.

The chimpanzee experiment claims that the apes that adopted a symbolic approach required more training time and made more errors during training, but once they had crossed the symbolic threshold, they were able to produce better sentences and learn new symbols faster. Figure 4.3 shows that this also holds for the simulated interpreters. Requiring less time to learn the first objects, the indexical learning curve grows steeper than the symbolic learning curve in the long run.

4.5 Discussion

We have set up the neural network experiments in order to investigate the differences between indexical and symbolic learning. Although such a difference can be shown to exist in our models, the experimental findings do not prove the accuracy of the models used nor do they validate the conversion from the chimpanzee language training program to the simulation. It should be noted that too many simplifications and assumptions had to be made to call these networks either indexical or symbolic interpreters. In order to reduce the complexity and tractability of the learning task, a relatively straightforward neural network is used. Also, even though a bias is avoided by adding the domain task, it is unclear how exactly the learning curve of the domain task and the symbolic curve ought to compare to the results of the indexical task. One should therefore be prudent with generalizing the particular model and approach used in these experiments.

However, when the results are projected onto the semiotic theory they allow for interesting conclusions to be drawn. The learning curves may help identify the mechanisms that underlie the shift to symbolism. The findings show that this shift may serve a practical purpose as it allows the subject to offload memory from one domain to another, thereby avoiding duplication of information. With selection pressure favoring language use, this could give an advantage to symbolic over non-symbolic systems. The findings also indicate that for a symbolic shift to take place in this model, the different domains (e.g. the domain of objects and actions and the domain of lexigram relations) are required to be mapped onto each other by the interpreter. Understanding how this mapping takes place is an important step toward a more accurate simulation of the interpretation process and the role of symbols herein.

Recall our two definitions of symbols, S1 and S2. In the case of S1, a lexigram would point directly to a referent (i.e. an index). According to the second definition S2, the symbol would also have a pointing relation to its referent, albeit a more obscure one which is embedded in a web of interdependent referential relationships. In the chimp experiments, the relations that exist among objects and lexigrams are also embedded in a web that spans both the lexigram domain and the object-action domain. A lexigram can be an index for another lexigram: their simultaneous use will likely lead to a pointing relationship from one to another (banana lexigram is usually followed by give lexigram, hardly ever by pour lexigram). The realm of objects and actions has a similar system of pointing relations (coke is always poured and never given). Therefore, a symbolic relation is, as one might say, a higher-order pointing relation from one domain to another. For the interpreter to create this kind of relation, it needs to find domains that can be mapped onto one another. Not every pairing of indexical systems is viable, there has to be a correlation between them that makes linking them purposeful. The input data presented in the symbolic task has some redundancy in it, so it makes sense for the interpreter to correlate the system of lexigrams with the system of actions and objects (table 4.3). It is exactly this redundancy or system iconicity (redundancy implies a lack of difference) in the topology of the systems that makes a symbolic relation advantageous [34]. A symbol, therefore, is a triadic relation that requires two systems of indices with topological redundancy, resulting in a higher-order index between two loci in those systems. The recognition of this redundancy, the insight that two domains are alike, is prerequisite for the symbolic shift to occur in an individual.

We can take this deconstruction of the sign one step further and consider what an index, being the constituent of symbols, is itself composed of. A pointing relation always points from one thing A to another B, which may in turn point to a third C and so on. The index from A to B is activated by the recognition of A (which is an iconic process). By virtue of their indexical relationship, A causes B to become active (as though B has been recognized). Suppose for example that A is smoke and B is a fire. The thought of a fire may cause a new thought C, no matter whether the fire was perceived directly (icon) or thought of after perceiving smoke (index). Consequently, what is caused by an index is also an icon.

The pointing relation itself is caused by a recurring appearance of signal and referent, being in close proximity to each other in one or more dimensions (i.e. spatial or temporal). Recognizing B frequently after recognizing A causes the interpreter to make a prediction about the future occurrences of B after A. The commonality of these situations is the simultaneous occurrence of signal and referent. Once the signal appears again, the interpreter recognizes the state as one of those situations where both signal and referent occur together. This recognition is itself a higher-order icon, because it classifies the signal-referent relation as one of many that have occurred before. Hence, an index is a relation between two icons that exists by virtue of a higher-order icon: their regular co-occurrence.

As an index is solely composed of icons, and a symbol is a particular configuration of indices, it follows that icons are the primary building blocks for all three types of interpretation. This conclusion does not imply that every iconic interpreter is also an indexical or symbolic interpreter. As the ape language training tasks as well as the simulation experiments show, a specific configuration is required for symbolic interpretation. Some apes were clearly unable to do symbolic interpretation even though they had indexical capacities. The neural networks that were trained to learn indices clearly show a behavior that differs from symbolic networks. Likewise, indexical interpretation requires a specific setup of iconic skills in order to induce the formation of a higher-order icon.

This conclusion would imply that iconic interpretation is a fundamental skill for interpretation. The firstness of icons is argued for in semiotics [124] but also by the proficiency of simple neural network models in classification tasks, where their robustness allows them to deal with distorted data [75, 97]. The potential of these computational models for recognition and classification tasks makes them a good starting point for further investigations into indexical and symbolic models of interpretation.

5

COMPUTATION, CREATIVITY AND CONSTRAINT

This short chapter[1] describes why a system needs to be capable of producing its own constraints in order to be self-programming. This contrasts with the idea that what computer programs produce is already, implicitly, present in their initial set of instructions. A capacity for transformational creativity turns out to be a crucial factor for systems to be considered self-programming.

5.1 Introduction

What causes a program's execution? Is it the design of the program, or the system on which it runs? In order for a program to execute as intended many times over, a stable environment is required in which instructions are flawlessly converted into physical operations. Computers take care of exactly that: they have been meticulously designed to maintain stability against noise and interaction, except for the operations of the program they provide an environment for. Systems have a natural tendency to degrade and fall apart; computers serve to constrain the physical surroundings of

[1] This chapter is based on [103] Leijnen, S. (2011). Thinking Outside the Box: Creativity in Self-Programming Systems. *Self-Programming in AGI Systems Workshop, August 4, 2011, Mountain View, CA.*

a program. They are the embodiment of a perfect unnatural mechanism, not only faster and more precise than the human mind, but also of a kind of dullness that we humans aren't capable of sustaining for a long period of time [16]. A program, then, is a set of ordered instructions that further constrains the executions that are possible on a computer. In order to guarantee that the execution goes as intended by the programmer, the program should be in a maximally constrained state, that is, it forces the computer to follow a single execution trajectory while making all other trajectories (e.g. those classified as logical errors, bugs or hardware failures) extremely unlikely to occur.

If programs are essentially ordered sets of instructions, then for a system to be self-programming it has to be capable of producing its own instructions. That is, new constraints that affect its operation. However, as a program is generally designed to be maximally constrained system, these changes are necessarily already part of the original program. A paradox resides in the theoretical concept of a self-programming algorithm: either the new constraint is already present in the program, and therefore not new, or it is created by the program itself - in which case it is not a constraint since the system could not be constrained any further. One way out of this paradox deals with creating programs that are not maximally constrained.

5.2 Combination, Exploration and Transformation

The ways in which constraint can be removed corresponds to the kind of creativity involved in the process. When dealing with computational creativity, a distinction is often made between combinatorial, exploratory and transformational creativity [15]. Combinatorial creativity is the discovery of a statistically unusual occurrence; exploratory creativity is the discovery of a new idea that had been a possibility all along; and transformational creativity deals with finding ideas that had previously been thought impossible.

We argue that combinatorial and exploratory creativity can be modeled fairly well with a maximally constrained computer program. For example, when a program searches for an optimal solution by traversing a designed search space along a designed or randomly generated path, it can be said to be capable of discovering statistically

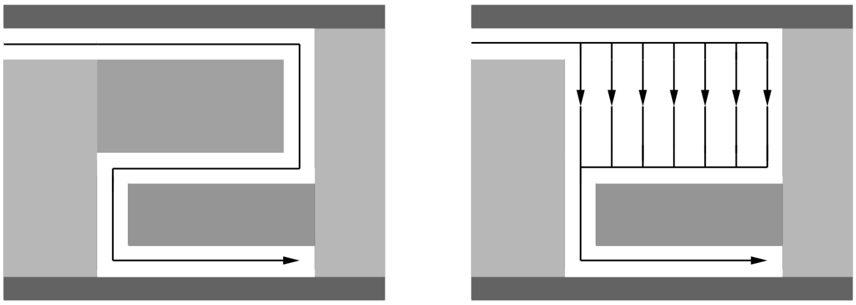


Figure 5.1: Schematic illustration of constraints affecting the execution of a program. Maximized constraint (left) and lowered constraint (right).

unusual occurrences, as well as discovering new ideas - albeit capable of finding only those allowed for by the search space.

Transformational creativity, however, cannot be modeled computationally in a straightforward way, as traversing a designed search space does not qualify as inventing something radically new. Ultimately, this search space is determined by a human programmer. The next section argues that, even if a program transforms the search space, this new space is implicitly present in the original program and therefore does not contain 'radically new' ideas.

5.3 Self-Programming Constraint

By definition, a maximally constrained program explores a single trajectory (figure 5.1, left). That is, the constraints of the search space are implicit in the constraints of the program. No constraints can exist that are not already implicitly present within the original constraints of the program; therefore, such a program cannot be considered exhibit transformational creativity or self-programming. This agrees with the notion that for transformational creativity to occur, partial independence from intentional control is a prerequisite [99].

Conversely, if a program is less-than-maximally constrained (i.e. due to causes that interfere with a program's execution), it would leave room for creating new constraints. Examples of these causes include programming errors, deviations in the physical embodiment, interaction by users or other programs, or (pseudo-)random functions. Such factors have in common that they can be noisy and unpredictable, features which can be employed to lower the amount of constraint on a program's execution. Consequently, instead of a constricted path there now exists a space of possibilities in which multiple paths are available, depending on the cause of interference and how it affects the program (figure 5.1, right).

Such a deconstrained space allows for the system to self-program new constraints. New, because the resulting path is not completely determined by the algorithm, but also by how the space is transformed by the interplay between program and the deconstraining interference. What appears to be noise disturbing the computational process, actually provides a breakdown of constraint that is necessary for transformational creativity and self-programming to occur [37].

5.4 Conclusions

A maximally constrained program uses exploratory creativity to change its state. For a system to be considered as self-programming, it needs to be capable of transformational creativity, which in turn requires a lessening of constraint in order for new constraint to come about. This is established by allowing for deviations from the maximally constrained path that the algorithm might follow, which are henceforth not part of the program itself.

In the end, merely allowing for random changes to the search space of a program is not a sufficient condition for qualifying it as self-programming; the search space also needs to change in a meaningful way. Creativity, contrary to random exploration, requires purpose [76]. So while bugs, errors and random numbers provide a means to get out of a designed search space, they do not specify where to get to, or how to get there. Another worthwhile observation is that the reduction of constraint causes a system's operations to become less well-defined. This makes the system lose some of the qualities that we tend to attribute to a computer program [96]. As a system becomes more self-programming, it becomes less of a program itself.

6

CONSTRAINT, SELF-ORGANIZATION AND AUTOGENESIS

The findings reported in the two previous chapters have launched an investigation into the nature of constraint. Origin of life theories often argue that molecular self-organization explains the spontaneous emergence of structural and dynamical constraints. However, the preservation of these constraints over time is not well-explained because of the self-undermining and self-limiting nature of these same processes. A process called autogenesis has been proposed in which a synergetic coupling between self-organized processes preserves the constraints thereby accumulated. This chapter[1] presents a computer simulation of this process (the Autogenic Automaton) and compares its behavior to the same self-organizing processes when uncoupled. We demonstrate that this coupling produces a second order constraint that can both resist dissipation and become replicated in new substrates over time.

[1]This chapter was submitted to Artificial Life Journal as [109] Leijnen, S., Heskes, T. and Deacon, T. W. (under review) Exploring Constraint: Simulating Self-Organization and Autogenesis in the Autogenic Automaton.

6.1 Introduction

How could life have emerged? Although understanding life's origins does not necessarily imply understanding life as we find it around us today [25], it promises far reaching consequences for scientific fields such as cognitive science [156] and artificial intelligence [102] as the emergent dynamics that created life may have also shaped mind, machine and society.

Theories explaining the possible origins of life are as numerous as the properties that are claimed to define it. Some define life based on a general capacity for replication [163] or the exchange of energy and matter with an external environment [113, 133]. Others characterize life as systems using template-based encoding (e.g. RNA) [90] or being composed of autocatalytic sets [125]. Although each of these theories has its own flavor in explaining life's possible origins [93], many assume the involvement of the structure forming dynamics of self-organization.

Despite their often pivotal role in explaining the emergence of life, self-organizing processes are limited in their capacity to maintain structure [129]. A recently proposed theory suggests that, beyond mere self-organization, a synergetic coupling between self-organizing processes is a minimal requirement for life [39]. Through this higher-order linkage, the processes that generate structure may persistently recreate a capacity for self-creation, leading to robustness and a potential capacity for long-term sustenance and natural selection. As an example of such an *autogenic* process, a proto-life model system called the autogen is described to illustrate how two self-organizing processes – reciprocal catalysis and self-assembly – maintain each other's boundary conditions and thereby mutually increase the probability of persistence over time [38].

Currently, the autogen model is a theoretical proposition that remains to be validated experimentally. As a step toward validation, this chapter describes a series of simulation experiments that investigate the self-organizing and autogenic properties of the autogen model. A simplified particle system simulation called the Autogenic Automaton models the synergetic linkage of self-organizing processes that leads to the emergence of autogens. In the next section, this higher-order linkage is described in more detail, followed by a description of the computational model and the methods used to simulate and quantify the self-organizing and autogenic processes that generate, eliminate, and preserve constraint.

6.2 Autogenesis

Life's ability to resist degradation and persist in hostile environments is both ubiquitous and astonishing. Generation of structure, preservation by repair, and trait persistence through reproduction are perpetually organized in a continuous struggle against the destabilizing mechanisms of the second law of thermodynamics. In this section, autogenesis is introduced in three parts: first, how the inevitable increase of thermodynamic entropy poses a problem for models of life based on self-organization. Then autogenesis is presented as a three-tiered process hierarchy of constraint elimination, constraint generation and constraint preservation (and ensuing constraint selection), after which this hierarchy is exemplified by the autogen model, to be used for the subsequent simulation experiments.

6.2.1 Second-Order Self-Organization

Self-organization may occur in open systems that are continually perturbed toward a far-from-equilibrium state through incessant nonlinear amplification of local fluctuations [129]. These systems tend to reduce their statistical entropy, i.e. their variety of potential states, thereby becoming statistically less complex; for this reason, self-organization may also be regarded as self-simplification [4].

A typical example is the formation of Rayleigh–Bénard convection cells [64], which may emerge if a fluid is heated from below, causing fast-moving molecules in the bottom to rise upward while slow molecules simultaneously move downwards. These two vertical motions lead to horizontal heat exchange between upward and downward moving molecules, obstructing the dissipation of heat from its source to the surface. Under certain conditions (e.g. temperature, viscosity, shape and size of the surface) hexagonal convection cells develop that minimize the horizontal heat exchange and thereby maximize vertical heat dissipation. As the number of potential system states is reduced by the emergence of these cells, the system becomes more ordered. Other examples include laser beams [73], where optical amplification results in spatial coherence; vortices, such as whirlpools and tornadoes; and autocatalytic chemical reactions, where the product of a reaction is also a catalyst for it, leading to a nonlinear reaction increase under particular proximity conditions [161].

The nonlinear amplification that is typical for self-organization tends to push the thermodynamic conditions for further propagation toward the unfavorable. This may occur up to a point where the system is no longer far-from-equilibrium and the local thermodynamic entropy increase comes to a halt. For example, in a reciprocally catalytic set, reaction rates may increase exponentially as more and more catalysts are produced, up until the point when not enough reactants are available for further propagation, and the self-organizing process ends. Given the universal presence of self-organization in living systems, how can it be possible that order persists long enough for complex organisms to come about? It has been suggested that the answer may lie in the hierarchical organization of self-organizing processes [39].

When the product of an autocatalytic reaction enables a second autocatalytic reaction, which produces a reactant that enables the first (or a third, etc., as long as the causal chain is eventually closed), a so-called *hypercycle* emerges [46]. Hypercycles represent one possible way in which self-organizing processes, autocatalytic cycles in this case, may be linked together in a dynamical process hierarchy. However, with respect to preventing self-undermining, this particular type of second-order self-organization does not provide a sufficient solution: each autocatalytic cycle that the hypercycle consists of represents a potential weakest link, which may cause the fragile hypercycle to break down entirely when reactants or energy for this particular cycle are no longer available.

Autogenesis requires another type of second-order self-organization where two (or more) self-organizing processes not only promote each other, but also where they act as a supportive environment if one of them breaks down, such that their self-undermining tendencies are reciprocally counteracted.

6.2.2 Constraint

The formation of crystals through self-assembly is a self-organizing process. The probability of particle detachment decreases with the number of adjacent crystal cells that keep a particle in place, further limiting diffusion of these particles and thereby reducing the uniformity of their spatial distribution (i.e. at some locations there are many particles present while at other locations there are few particles to none). The attachment and detachment rates of the assembly process, together with the particular binding properties of the particles that make up these crystals, generate a constraint on the spatial distribution of particles. More generally, a reduction of the variety of macroscopic states can be understood as a constraint producing process [37].

From a constraint-centric perspective, the global increase of entropy predicted by the second law of thermodynamics appears to run counter to the production of constraint caused by self-organization. Thermodynamic entropy increase spontaneously introduces noise into the system, as the probability of a random microscopic event inducing more order is lower than the probability of an event inducing less order (there are relatively few ordered states), thereby increasing the variety of macroscopic states. An example of such a constraint eliminating is the noise inducing process of an ice cube melting in a soda drink: considering the possible states of water molecules in the glass, the number of states where only some of the molecules are arranged in a solid ice-cube is by far outnumbered by the number of states where they are all mixed up.

Living systems tend to produce constraint as well as preserve it against elimination. This capacity allows organisms, and lineages of organisms, to persist over long stretches of time. Following the type of second-order self-organization described above as autogenesis, constraint preservation is enabled by a juxtaposition of constraint producing processes, such that they actively support each other's persistence [35]. Whereas self-promoting self-organizing processes (e.g. hypercycles) tend toward self-undermining and ultimately a breakdown of the causal cycle, this reciprocally counteracting juxtaposition actively prevents self-undermining from taking place.

The relatively stability of these structural synergies allows for a simple type of natural selection to occur, as different kinds of synergies may co-exist within the same system. Some will be better suited to prevailing conditions than others and therefore have a better chance of sustaining themselves. This eventually leads to an elimination of noise (or reduction of variety) on a higher level, as unsuccessful noise-reducing synergies are removed. In this higher-order selection dynamic, a discontinuity becomes apparent again, as the structured parts are separated in a competition caused by the dissipative potential of the whole. This transition represents a shift in logic comparable but opposite to the emergence of ordered structure from chaos: it could be argued that this shift constitutes a secondary kind of emergent transition between dynamical regimes.

6.2.3 Autogen

The three-tiered constraint hierarchy of autogenesis is exemplified by a minimal model system of life, the so-called *autogen*. In earlier publications (e.g. [38]), this model system was referred to as 'autocell'. However, since it exemplifies a general class of self-generating systems (rather than being typically cellular) 'autogen' is the preferred term. Autogens are formed by a synergistic relation between self-assembly and reciprocal catalysis, both being self-organizing processes.

In a reciprocally catalytic system, each reaction initially leads to an increased probability for another reaction to take place, as more and more catalysts are created. Exponential growth ensues until the reactants are depleted. Reciprocal catalysis leads to exponential increase of reactions that is limited solely by the number of available reactants. A boundary or container would prevent the exhaustion of reactants by removing them from the environment, thereby preserving a chemical potential for further dissipation [113]. Such a container may itself be formed by a self-organizing process, e.g. crystal growth through self-assembly [48]. Autogenesis, then, suggests that the form and function of a self-assembled container is dynamically linked to the autocatalytic process as it prevents the reactants from being depleted. Similarly, it explains how the autocatalytic process dynamically shapes the form and function of the crystals as it affects the process of self-assembly.

The autogen model is illustrated in figure 6.1. Particles of type A and B react to form a C particle, catalyzed by particles of type F. A similar reaction takes place for particles of type D and E, which form F and G particles, catalyzed by C particles. In the same system, G particles attach to one another creating self-assembled crystals of type G^n, with n for the number of G particles the crystal consists of. Due to their particular form, these crystals may contain catalysts, thereby negatively affecting the immediate production of G particles but ensuring a potential for G-particle production over time. The negative part of the structural coupling ensures that self-assembly stops before the catalysts are depleted, even though they are contained and therefore not readily available in the environment. This reaction potential is employed when a crystal opens up after detachment: contained catalysts are released, initiating a new chain of catalytic reactions that provides new G particles used to repair the container, after which it may close again, thereby completing the work cycle.

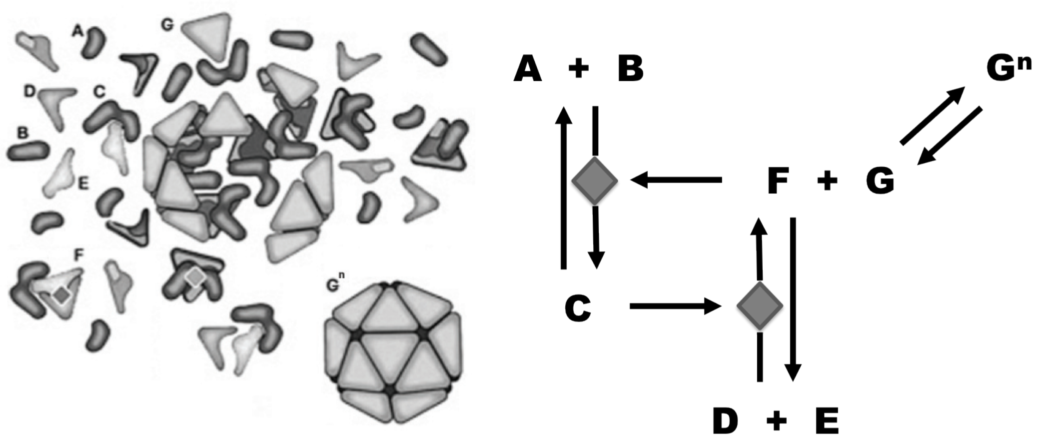


Figure 6.1: Illustration of autogen formation (left) and the reactions involved (right). Autogens constitute a dynamical linkage between self-assembly and reciprocal catalysis. In this model system, self-assembly is a self-organizing process where G particles attach to one another, forming G^n crystals of size n. These crystals may break up due to detachment. G particles are generated by a reciprocally catalytic set of six different particle types (A to F). In turn, crystals may contain C and F particles, isolating these catalysts from potential reactants.

The autogenic process as a whole then gains a minimal form of autonomy: as it is able to do work on its own conditions for sustenance, it grows independent from the conditions of its environment and becomes more dependent on its own internalized constraint. Under some conditions that are strived to be maintained, probability of growth and sustainment is higher than that of breakdown. When the autogen is damaged, it likely begins to repair itself. Autogenesis is about the higher-order constraint on the constraint generating processes of which it is constituted, such that a self is reproduced. It is a dynamic for the maintenance of itself as it maintains an implicit description of its units of preservation.

6.3 The Autogenic Automaton

The logical steps building up to autogenesis, exemplified by the autogen model described above, are simulated with the Autogenic Automaton. Other computational proto-life models have simulated the formation of containers, the emergence of collectively autocatalystic sets, or both [115, 162]. Although this simulation falls into the latter category, the goal here is not to provide a physically accurate model of either self-organizing process, nor of their synergy, but rather to demonstrate the viability of the logical hierarchy leading up to this synergy.

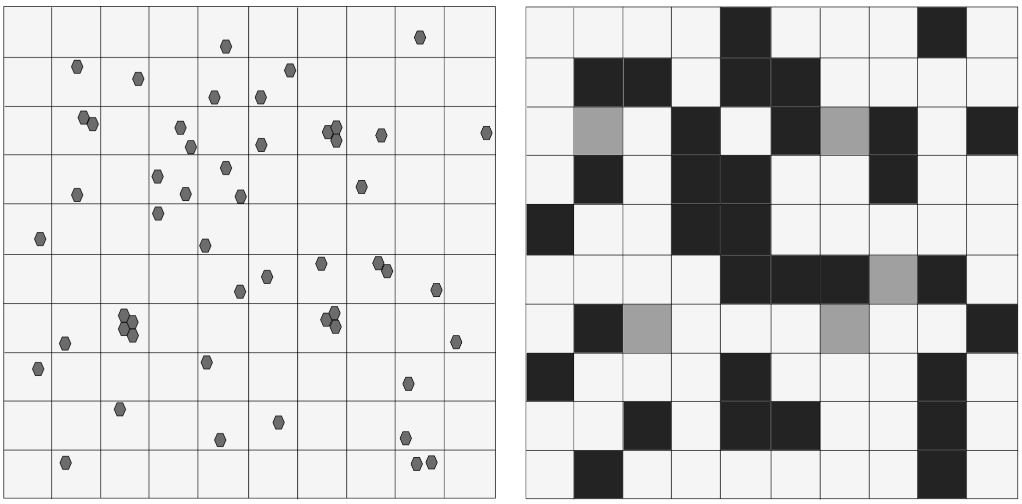


Figure 6.2: A continuous, closed particle system (left) is modeled in the Autogenic Automaton as a discrete grid of 10 x 10 tiles (right).

6.3.1 Model

A two-dimensional 10 x 10 tile grid is used as a discrete model of a closed reaction-diffusion system (figure 6.2). Particle movement and the reaction rules that govern particle attachment, detachment, and the creation and removal of particles are all computed locally per tile. In this sense, the system resembles a cellular automaton allowing for emergence [11], where tiles may be occupied by multiple particles. Diffusion caused by particle-to-particle collisions is approximated by randomly directed movement to a horizontally or vertically neighboring tile with probability $P_{\text{movement}} = 0.1$

at every time step, for each particle.

Particle-to-particle reactions and associated probabilities are modeled after crystallization and reciprocal catalysis. To this end, six particle types are defined and used for modeling reciprocal catalysis while a seventh type models the formation of crystals through self-assembly. Further implementation details, including the reaction rules, are given in section 6.9.

6.3.2 Simulation

The Autogenic Automaton is initialized by assigning random grid locations to a predefined quantity of particles of each type. Next, for a given number of time steps, new particle positions are computed and the reaction rules are applied to the particles at each tile.

One advantage of these localized reactions is that only small subsets of the total number of particles interact at each time step, reducing the computational complexity of the simulation. Another way to keep the model relatively simple and the simulation computationally tractable is to model only the aspects of self-organization that are necessary for showing the viability of autogenesis, which include nonlinear probability functions and reversible reactions. Other physical properties typical for particle systems (e.g. kinetic energy, dissipation of heat) are not modeled.

In order to show the generation and elimination of constraint (i.e. macroscopic change) over time, the simulation starts in non-equilibrium conditions. Due to the absence of heat and friction, the entropy potential necessary for far-from-equilibrium systems to be self-organizing is not defined with respect to thermodynamic equilibrium (when all movement and reactions have ceased to occur) but rather with respect to chemical equilibrium. Thus, the initial set of particles is not in chemical equilibrium since no crystallization has occurred nor has a catalytic reaction taken place. So, the macroscopic change that is observed in the experiments that follow may be attributed to the system moving toward a chemical equilibrium.

6.3.3 Quantifying Constraint

Through the course of a simulation run, particles move and collide against one another to create new particles or crystals, or they fall apart. The system moves through various macroscopic states, caused or maintained by processes that generate, eliminate, preserve or select constraint; the quantification of statistical entropy described here yields an indirect observation of the underlying processes.

Constraint is expressed by means of the information entropy over the spatial probability distribution of particular event types in the tile grid [74, 92, 127]. Notable characteristics of the particle system surface by observing the (in-)homogeneity of the locations of events that correspond to these characteristics, such as the presence of a particle type or the occurrence of a specific reaction.

Given a set of probabilities p_i for $i = 1, \ldots, n_{\text{tiles}}$ with $0 \leq p_i \leq 1$ and $\sum_i p_i = 1$, information entropy is defined as [32, 141]:

$$H = -\sum_{i=1}^{n_{\text{tiles}}} p_i \log_2 p_i \, .$$

Now, when we consider an event X, we substitute

$$p_i = \frac{|X_i|}{|X|} \, ,$$

with $|X_i|$ the number of events at tile i and $|X| = \sum_i |X_i|$ the total number of events, to obtain

$$H(X) = -\sum_{i=1}^{n_{\text{tiles}}} \frac{|X_i|}{|X|} \log_2 \frac{|X_i|}{|X|} \, .$$

For ease of interpretation we often consider the so-called normalized information entropy [89]

$$\hat{H}(X) = -\frac{1}{\log_2 n_{\text{tiles}}} \sum_{i=1}^{n_{\text{tiles}}} \frac{|X_i|}{|X|} \log_2 \frac{|X_i|}{|X|} \, , \tag{6.1}$$

which normalizes the standard information entropy by its maximum value such that always $0 \leq \hat{H}(X) \leq 1$. For a completely homogeneous distribution of events over tiles we now have $\hat{H}(X) = 1$, whereas $\hat{H}(X) = 0$ when all events X are concentrated at a single tile.

6.3.4 Quantifying Multiple Constraint Types

Where it is necessary to consider the interaction between two dynamical processes, the Kullback-Leibler divergence is used [100]:

$$D_{KL}(P||Q) = \sum_{i=1}^{n_{\text{tiles}}} p_i \log_2 \frac{p_i}{q_i} \, .$$

Given two event types X and Y, their co-location in the tile grid gives an indirect measure of this interaction. Substituting p_i and q_i with $\frac{|X_i|}{|X|}$ and $\frac{|Y_i|}{|Y|}$, respectively, would yield infinite divergence for a distribution with a tile i such that $|X_i| > 0$ and $|Y_i| = 0$. To resolve this problem, a smoothing function is used [10], where

$$q_i = \begin{cases} \alpha \frac{|Y_i|}{|Y|} & \text{for} \quad |Y_i| > 0 \\ \epsilon & \text{for} \quad |Y_i| = 0 \end{cases} \, , \tag{6.2}$$

with $\epsilon = 10^{-5}$ and normalization coefficient α chosen such that the probabilities sum to 1. p_i is substituted similarly with $\frac{|X_i|}{|X|}$. For ease of exposition, we will omit this smoothing in subsequent formulas.

6.4 Constraint Generation

The Autogenic Automaton is used to simulate the generation of constraint in two separate self-organizing processes: the formation of crystals through self-assembly, and the local amplification of reactions taking place in a reciprocally catalytic set.

6.4.1 Self-Assembly

Self-assembly is modeled by a series of attachment and detachment reactions between G particles and crystals G^n with reaction parameters γ^+ and γ^-:

$$G^n + G \rightleftarrows G^{n+1} \, , \tag{6.3}$$

with $n \geq 1$ and $G^1 \equiv G$. If a G particle is located within the same tile as either another G particle or crystal, the probability of attachment is given by

$$P_g^+ = \gamma^+ \in [0, 1] \, .$$

Modeling detachment as the opposite of attachment, crystals have a probability P_g^- of a G particle detaching from the crystal. Larger crystals are more tightly connected and less likely to break apart than smaller crystals due to a larger number of kinks holding the individual particles together [21]. An increased size yields a lower probability of detachment and therefore increases the probability for further growth. This introduces a nonlinearity in the crystal growth process, reflected in our model system by a detachment probability function that is negatively exponential to the crystal size n.

$$P_g^- = (1 + \exp[\gamma^-])^{-n} ,$$

with $\gamma^- \in \mathbb{R}$, $n \geq 2$. Following equation (6.1), event X_i is defined with respect to self-assembly as the observation of a G particle at tile i, where G^n crystals are counted as n observations. Therefore, the generation of constraint during this process is examined using

$$\hat{H}(G,t) = -\frac{1}{\log_2 n_{\text{tiles}}} \sum_{i=1}^{n_{\text{tiles}}} \frac{|G_i(t)|}{|G(t)|} \log_2 \frac{|G_i(t)|}{|G(t)|} ,$$

for the normalized information entropy of G at time t. Figure 6.3 shows the development of $\hat{H}(G,t)$ over time, for different values of γ^-. With $\gamma^- = -5$, the probability of detachment P_g^- is relatively high, such that many crystals fall apart and the constraint on particle G locations is low. With $\gamma^- \in [-1, -2]$, P_g^- is relatively low: once formed, crystals do not break apart, leaving no single G particles for attachment and no room for further growth. The G particle locations are maximally constrained for $\gamma^- = -4$.

6.4.2 Reciprocal catalysis

Particle types A to F are used to model self-organization through reciprocal catalysis. Particles of type A and type B may react to form a C particle when both are located in the same tile; similarly for particles D and E forming an F:

$$A + B \overset{F}{\rightleftarrows} C , \tag{6.4}$$

$$D + E \overset{C}{\rightleftarrows} F . \tag{6.5}$$

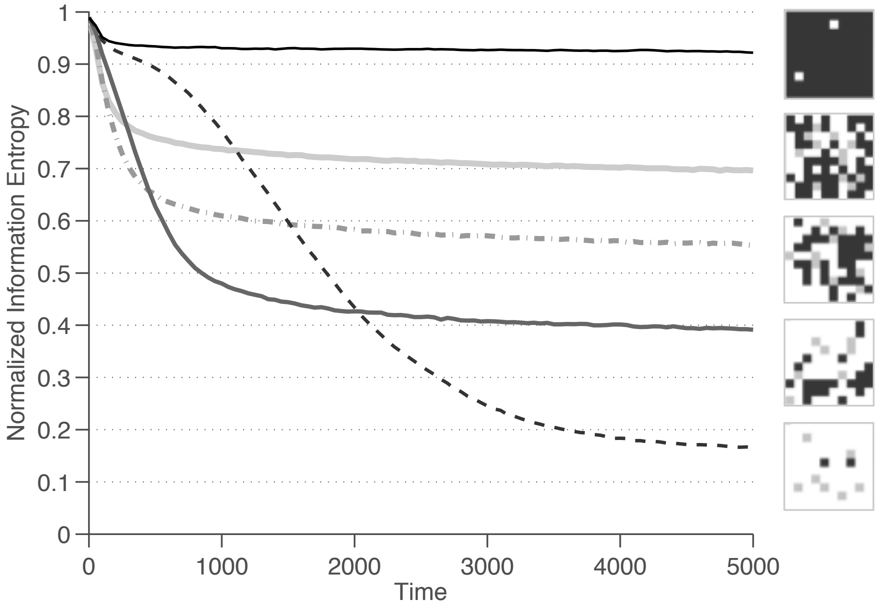


Figure 6.3: Decrease of normalized information entropy $\hat{H}(G,t)$ during self-assembly of 1000 G particles with $\gamma^+ = 1$, for several detachment probabilities: $\gamma^- = -5$, $\gamma^- = -4$, $\gamma^- = -3$, $\gamma^- = -2$, $\gamma^- = -1$. The images on the right depict how $\hat{H}(G,t)$ correlates with the distribution of G particles over the grid at the end of a run, ranging from an almost homogeneous distribution (top) to a few G^n crystals (bottom). Results are averaged over 100 trial runs.

Particles F and C are catalysts for reactions (6.4) and (6.5), respectively. Reaction probability P_r^+ increases exponentially with n, the number of catalysts present at the same tile i (i.e. $n = |F_i|$ for the former reaction, and $n = |C_i|$ for the latter)

$$P_r^+ = (1 + \exp[\varrho^+])^{-(1+n)^{-2}} ,$$

with $\varrho^+ \in \mathbb{R}$. The reverse reactions (C particles splitting into A and B particles and F into D and E) occurs with probability P_r^- for every C or F particle, at each time step

$$P_r^- = \varrho^- \in [0,1] .$$

Similar to self-assembly, reciprocal catalysis is a locally nonlinear process: one catalytic reaction increases the likelihood of another catalytic reaction occurring. However, the observable artifacts of reciprocal catalysis (i.e. the produced catalysts) may cease to exist once this amplification process no longer takes place, or they may diffuse to different locations. To quantify the amount of constraint generated, we therefore use the probability distribution of reaction locations as observed events, rather than the catalysts themselves.

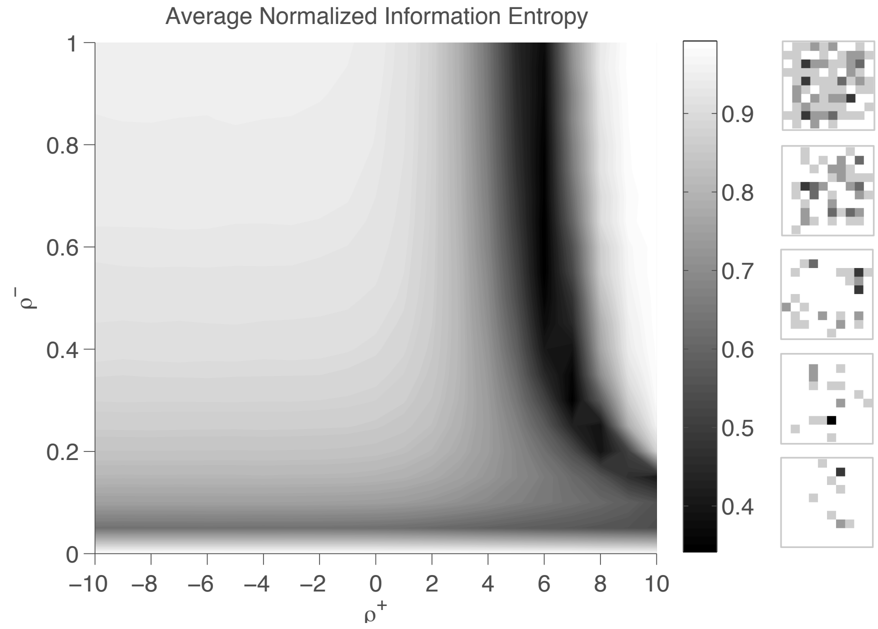


Figure 6.4: Normalized information entropy during reciprocal catalysis, after initialization with 1000 particles distributed equally among types A, B, D, and E. For given ϱ^+ and ϱ^-, $\hat{H}(R,t)$ is averaged over 5000 time steps and 10 trial runs. The right-side images depict the distribution of catalytic reactions over the grid at the end of a run.

The constraint generated by reciprocal catalysis is quantified by a decrease in normalized information entropy over the distribution of catalytic reaction loci R at time t:

$$\hat{H}(R,t) = -\frac{1}{\log_2 n_{\text{tiles}}} \sum_{i=1}^{n_{\text{tiles}}} \frac{|R_i(t)|}{|R(t)|} \log_2 \frac{|R_i(t)|}{|R(t)|} \, .$$

In order to investigate the effect of parameters ϱ^+ and ϱ^- on the normalized information entropy, $\hat{H}(R,t)$ is averaged over time:

$$\frac{1}{t_{\max}} \sum_{t=1}^{t_{\max}} \hat{H}(R,t) \, .$$

Results for $t_{\max} = 5000$ are shown in figure 6.4. We found that distribution R is most constrained for $\varrho^+ \approx 6$ and $\varrho^- > 0.5$ (i.e. catalysts break up regularly) .

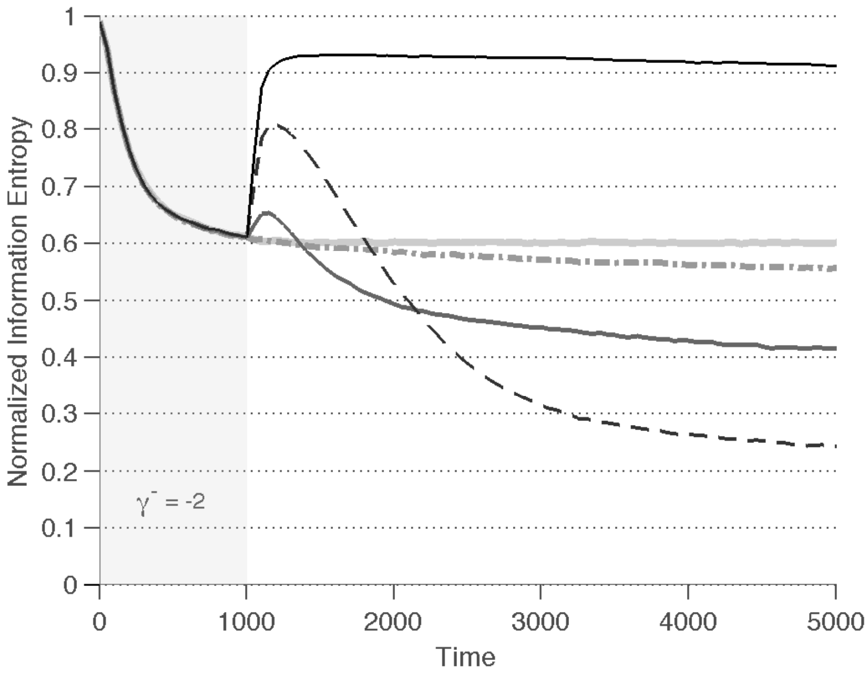


Figure 6.5: Elimination of constraint: the Autogenic Automaton is initialized with 1000 G particles and $\gamma^+ = 1$, $\gamma^- = -2$. At $t = 1000$, γ^- is changed: $\gamma^- = -5$, $\gamma^- = -4$, $\gamma^- = -3$, $\gamma^- = -2$, $\gamma^- = -1$. For $\gamma^- \in [-3, -4]$ the eventual decrease in information entropy is preceded by an initial increase. Results are averaged over 100 trials.

6.5 Constraint Elimination

Self-organizing processes are enabled by specific conditions that promote local amplifications. Due to change initiated by self-undermining (or imposed externally) the boundaries of these conditions may be transgressed leading to an elimination of previously generated constraint. This process can be demonstrated by means of a so-called *process spectrometry*: after an initial phase with conditions that enable self-assembly, the value of γ^- is changed, and the effect on the normalized information entropy is observed (figure 6.5).

The figure shows that after $t = 1000$ the normalized information entropy is (initially) higher for $\gamma^- \in [-5, -3]$. After some time this value decreases as constraint grows: G particles that have previously detached from small crystals now attach to others, such that only large crystals remain. Within the 5000 time steps shown, this effect results in higher local concentrations of G particles, less homogeneity, a lower value of $\hat{H}(G, t)$ and more constraint for $\gamma^- \in [-4, -3]$.

6.6 Constraint Preservation

The higher-order linkage between self-assembly and reciprocal catalysis has been discussed in section 6.2.3, where a mutually constraining coupling is suggested that works in two directions:

1. G particles generated by autocatalysis are created in close proximity to one another, due to the locality of the catalytic amplifications, thereby increasing the likelihood of crystal growth;
2. crystals may act as containers for catalysts, preventing the reciprocally catalytic process from undermining itself and preserving a potential for catalysis at a later point in time.

Modeling this synergetic linkage, reaction 6.5 is changed as follows:

$$D + E \xrightarrow{C} F + G\,.$$

This increases the likelihood of G particles being produced in close proximity to a catalytic reaction. For the reverse linkage, a crystal needs to be capable of containing catalysts upon formation, to be released again when the crystal breaks up. Therefore reaction 6.3 is changed, where a crystal of size n containing k C particles and m F particles is denoted as $G^n(kC, mF)$:

$$G^n(kC, mF) + G + pC + qF \rightarrow G^{n+1}((k+p)C, (m+q)F)\,,$$

$$G^n(kC, mF) \rightarrow G^{n-1} + G + kC + mF\,,$$

with $k, m, p, q \geq 0$ and $n \geq 2$. The continuous addition of G particles is balanced by removing G particles at each time step with probability $P_g^- = (1 + \exp[\gamma^-])^{-1}$.

6.6.1 Parameterization

Thus far, four reaction parameters have been introduced: γ^+, γ^-, ϱ^+ and ϱ^-. The results in section 6.4.1 showed that a decrease in $\hat{H}(G,t)$ may occur when γ^+ is fixed at 1. Similarly, figure 6.4 shows that $\varrho^- > 0.5$ allows for relatively low values of $\hat{H}(R,t)$. Given these values, we investigate the ranges of γ^- and ϱ^+ that allow for both types of self-organization to occur simultaneously.

The redundancy between the distributions of G particle locations and catalytic reactions is considered to be an indication of the amount of interaction between both processes, i.e. it measures whether crystals tend to be located in proximity to catalytic reactions, and vice versa. This is quantified using the Kullback-Leibler divergence with smoothing (eq. 6.2), which is symmetrized to obtain a commutative measure:

$$\begin{aligned} SD_{KL}(G,R,t) &= D_{KL}(G(t)||R(t)) + D_{KL}(R(t)||G(t)) \\ &= \sum_{i=1}^{n_{\text{tiles}}} \left[\frac{|G_i(t)|}{|G(t)|} - \frac{|R_i(t)|}{|R(t)|} \right] \log_2 \left[\frac{|G_i(t)|}{|G(t)|} \Big/ \frac{|R_i(t)|}{|R(t)|} \right] , \end{aligned}$$

with smoothing (eq. 6.2) applied if necessary. Running constraint preservation experiments requires parameterization of γ^- and ϱ^+ such that

(a) self-assembly takes place (i.e. $\hat{H}(G,t)$ is low);

(b) reciprocal catalysis takes place (i.e. $\hat{H}(R,t)$ is low);

(c) both processes take place in each other's proximity (i.e. $SD_{KL}(G,R,t)$ is low).

Figure 6.6 shows these three measurements averaged over 5000 time steps for different values of γ^- and ϱ^+. The desired parameter values are estimated by minimizing

$$\hat{H}(G,t) + \hat{H}(R,t) + \beta\, SD_{KL}(G,R,t) ,$$

where coefficient β scales $SD_{KL}(G,R,t)$ to $[0,2]$

$$\beta = 2 \left(\max_{\gamma^-,\, \varrho^+ \in [-10,10]} SD_{KL}(G,R,t) \right)^{-1} ,$$

such that the normalized information entropies and the divergence between the distributions contribute equally to the sum.

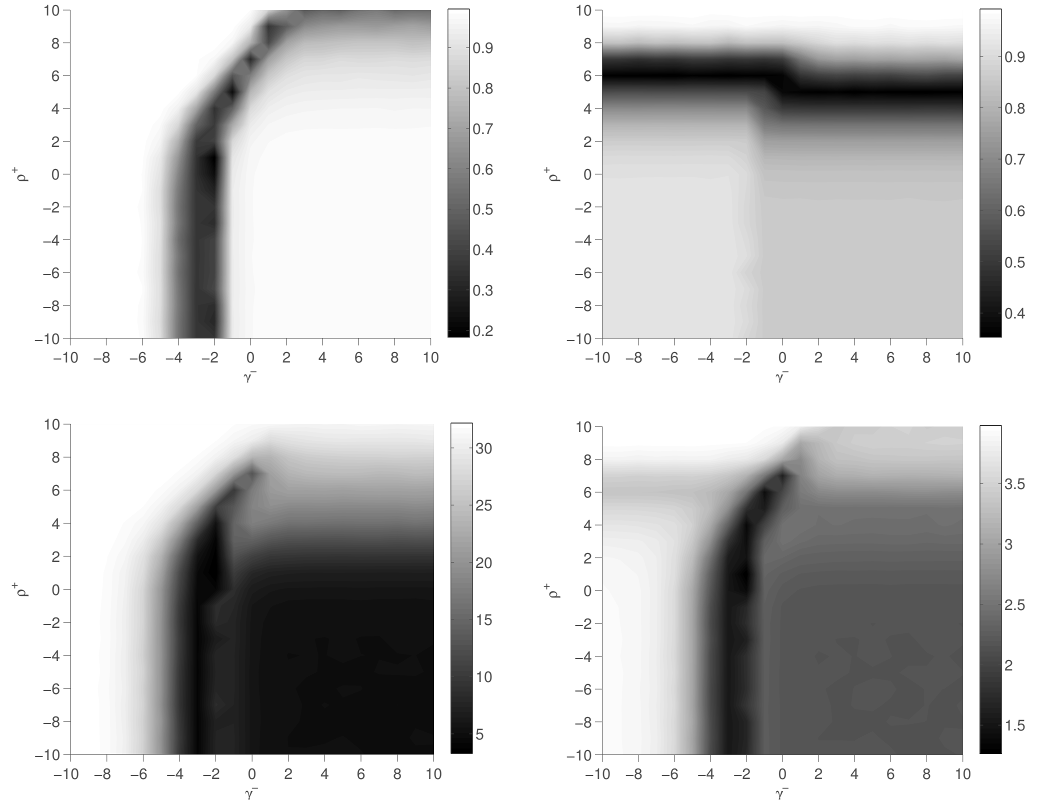


Figure 6.6: After initializing the simulation with 1000 particles evenly distributed among types A, B, D and E, it is run for 5000 time steps with $\gamma^+ = 1$, $\varrho^- = 0.5$ and $\gamma^-, \varrho^+ \in [-10, 10]$. The four figures above show the average normalized information entropy of G particle locations (top left), the average normalized information entropy of catalytic reaction locations (top right), the average symmetrized Kullback-Leibler divergence with smoothing, where $SD_{KL}(\gamma^-, \varrho^+) = \max(SD_{KL})$ if no occurrences are found (bottom left), and the sum of these three figures, where SD_{KL} has been normalized using scaling coefficient β (bottom right).

6.6.2 Preservation

Using the parameter values found, a process spectrometry is generated (figure 6.7). For $\gamma^- \in [-5, -4]$ after $t = 1000$, the high probability of detachment is not conducive to the prolonged persistence of crystals, and they fall apart. For γ^- remaining at 0, $\hat{H}(G, t)$ continues to develop unperturbed. Changing γ^- to -1 results in a lower normalized information entropy, as more G particles detach and subsequently attach to larger crystals (cf. figure 6.3). With γ^- changed to -2, $\hat{H}(G, t)$ initially drops, but eventually finds a new equilibrium at a higher value than with $\gamma^- \in [-1, 0]$.

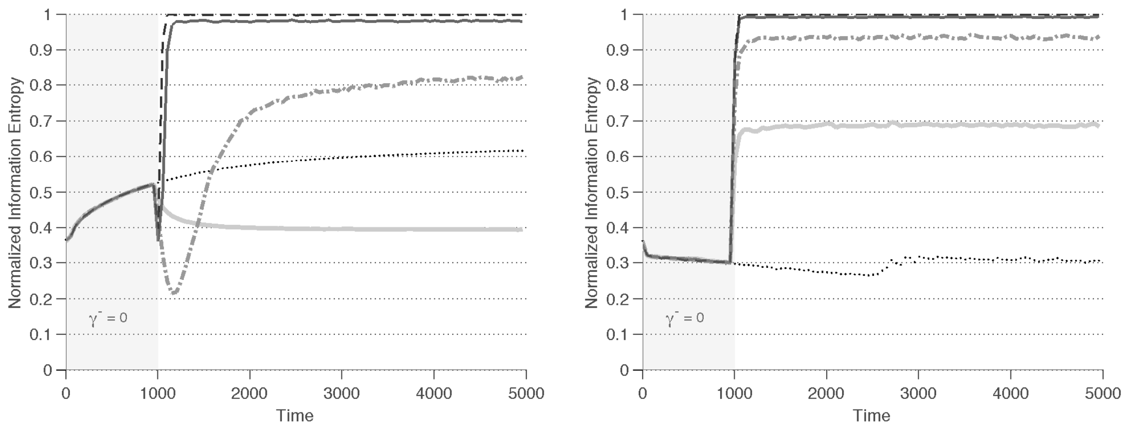


Figure 6.7: Normalized information entropy $\hat{H}(G,t)$ during autogenesis (left) and self-assembly without reciprocal catalysis where instead of being generated by a catalytic reaction, G particles are added to random grid locations with the same rate as in the previous simulation (right). With $\gamma^+ = 1, \varrho^+ = 6$, $\varrho^- = 0.5$, and $\gamma^- = 0$ for $t \in [0, 1000]$, and 1000 particles evenly distributed among types A, B, D and E initially. For $t \in [1001, 5000]$, $\gamma^- = -4$, $\gamma^- = -3$, $\gamma^- = -2$, $\gamma^- = -1$, $\gamma^- = 0$. Results are averaged over 1000 trials.

By itself, this figure provides limited insight into the processes that underlie the production, elimination and preservation of constraint. However, it may be compared against a similar experiment that lacks a higher-order linkage. Since $\hat{H}(G,t)$ only measures the constraint of the distribution of G particles, the experiment is repeated without the set of particles necessary for reciprocal catalysis. During the process spectrometry of figure 6.7 (left), the number of G particles that were added at each time step is stored. In this second experiment, G particles are created at random grid locations at exactly the same rate. This allows for self-assembly to take place under similar circumstances, but decoupled from reciprocal catalysis. G particle creation occurs at the same rate, although spatial proximity is no longer biased by reciprocal catalysis, and the absence of catalysts excludes the possibility of containment.

The results of this self-assembly experiment are shown in figure 6.7 (right). Comparing both figures, it is found for $\gamma^- = -2$ that constraint is preserved when a synergetic linkage between self-assembly and reciprocal catalysis is present, while it largely falls apart in the case of mere self-assembly. Also for $\gamma^- = -1$, the value of $\hat{H}(G,t)$ remains lower with this linkage than without. This is not the case when γ^- remains at 0. Here, self-assembly results in a more constrained system than when combined with reciprocal catalysis.

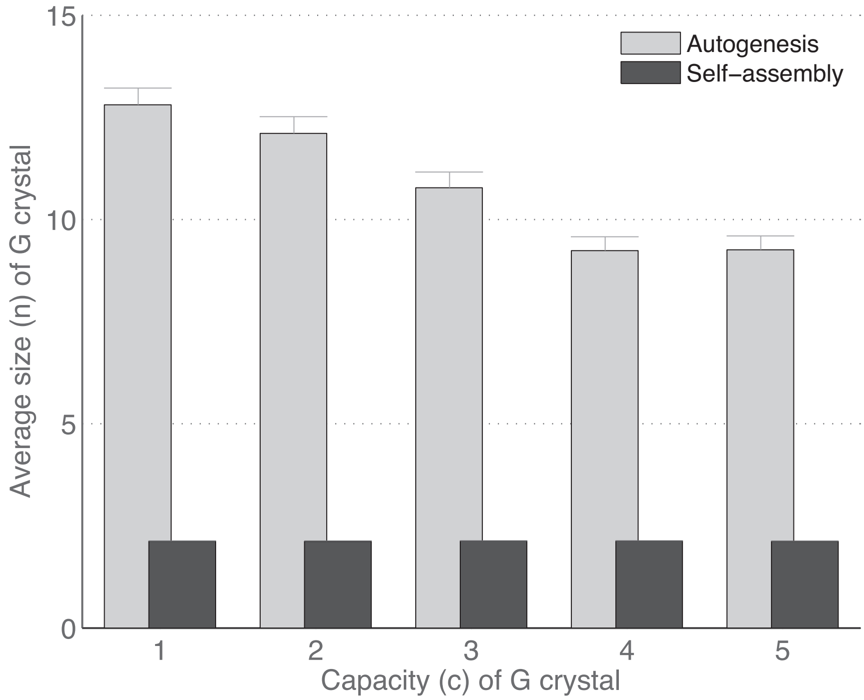


Figure 6.8: Fixing γ^- at -0.5 and $\gamma^+ = 1, \varrho^+ = 6$, and $\varrho^- = 0.5$, the grid is again initialized with 1000 particles of types A, B, D and E. For 1000 trial runs over 5000 time steps, the mean crystal size and standard error of the mean are reported for the five different containment capacities.

6.7 Constraint Selection

The preservation of constraint is a higher-order process: not only is noise reduced as the spatial distribution of events becomes more constrained, but the distribution over specific constraint types, e.g. the specific form of crystals, is itself also reduced.

Figure 6.9: Due to the way G particles attach to one another, crystals with specific topologies may come about. This illustration shows several crystals G_c^n of equal size (i.e. $n = 5$) but with different capacities (c) for containing catalysts, due their specific form.

In order to experimentally show the emergence of such higher-order constraint, the containment capacity of crystals is limited by their specific form (figure 6.9). This is reflected in the Autogenic Automaton by initializing each new crystal with property c, a random value ranging from 1 to 5 that indicates the maximum number of catalysts a crystal may contain.

Figure 6.8 shows the average size of crystals compared against their maximum containment capacity, for both autogenesis and self-assembly without reciprocal catalysis. The results reveal two differences. First, a difference in crystal size between autogenesis and self-assembly, which could have been inferred from the results of the previous constraint preservation experiments.

The second difference is that, for autogenesis, a crystal's containment capacity is correlated with its average size. In our model system, the removal probability P_g^- is independent of the specific crystal form, so the maximum containment of a crystal does not affect its size directly. Rather, the value of c indirectly affects the growth of crystals, as the numbers of catalysts present at a tile will affect the production of new G particles, and thereby a crystal's capacity for formation (or reconstitution) if those catalysts are released upon detachment. This work cycle creates a difference in size between the different crystal topologies, which is maintained despite the independence between crystal topology and the underlying self-organizing processes.

This higher-order constraint may be quantified using the normalized information entropy over the distribution of the containment capacities of crystals. With p_c the probability that a crystal has a containment capacity c, $|G_c^n|$ the number of crystals with capacity c and $|G^n| = \sum_c |G_c^n|$ the total number of crystals is

$$p_c = \frac{|G_c^n|}{|G^n|} ,$$

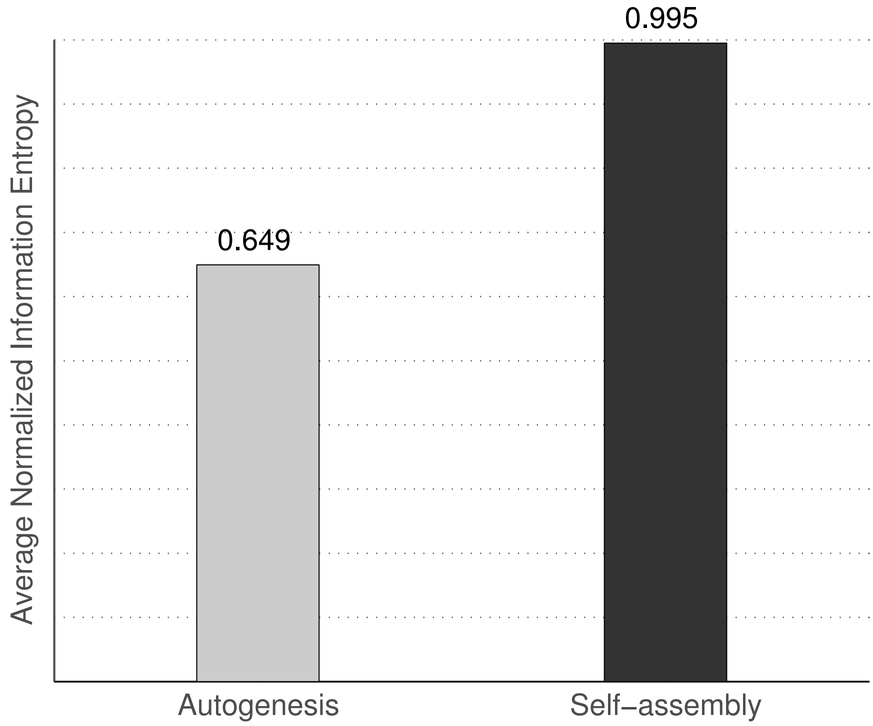


Figure 6.10: Higher-order constraint: the average normalized information entropy over the distribution of crystal capacities $\hat{H}(G_c^n, t)$ is substantially lower for autogenesis than for self-assembly without reciprocal catalysis. Results averaged over 1000 trial runs.

$$\hat{H}(G_c^n, t) = -\frac{1}{\log_2 5} \sum_{c=1}^{5} \frac{|G_c^n(t)|}{|G^n(t)|} \log_2 \frac{|G_c^n(t)|}{|G^n(t)|} .$$

Figure 6.10 shows the average $\hat{H}(G_c^n, t)$ over the simulation runs of the previous experiment for self-assembly and autogenesis. Selection of crystal topologies induced by specific conditions accounts for the relatively low normalized information entropy over containment capacities during autogenesis.

6.8 Conclusions

Constraint generation, elimination, preservation and selection have been shown to occur in the Autogenic Automaton simulation under particular conditions. Taken together, these processes constitute self-organization and autogenesis, albeit in a minimal sense. The self-undermining tendency of self-assembly and reciprocal catalysis is limited by virtue of their second-order synergy – a higher-order constraint on these constraint generating processes – leading to preservation of autogens and ultimately selection of crystal topology.

The experimental explorations described in this chapter are not intended to quantify autogenesis, or to give a full account of autogenic properties and phenomena. Rather, they serve to demonstrate (1) the dynamics reversal that takes place in second-order self-organization, and (2) the higher-order noise reduction constituted by formal type selection that may emerge from competition between simple autogens. Those properties are what sets autogenic systems apart from other models of the emergence of proto-life.

The hierarchical distinction between different types of constraint is reminiscent of a logical-type distinction. Although the three statistical distributions used here (G particle locations, R reaction locations, and G_c^n containment capacities) are all subject to quantification in terms of information entropy, this quantification does not distinguish between physico-chemical constraints expressed by the former two, and substrate independent, formal constraint expressed by the latter. Since the physics underlying the maintenance of far-from-equilibrium states have largely been ignored in this simulation (cf. [6]), further research and simulation is required to develop tools capable of expressing this dynamical difference [40].

6.9 Appendix: Model Description

A non-toroidal two-dimensional 10 x 10 tile grid is used as a discrete model of a reaction-diffusion system containing seven different types of particles (A to G). At the start of a simulation run, the particles are distributed randomly over the grid. Then, for each time step, each particle may

1. move to a neighboring tile;
2. collide and react with other particles in the same tile;
3. break up.

These steps are described in more detail below.

Movement

Random movement caused by bouncing against the edges of the grid, or particle-to-particle collision, is approximated by randomly directed movement to a horizontally or vertically neighboring tile with $P_{\text{movement}} = 0.1$ for each particle or crystal.

Collision

Self-assembly of G^n crystals:

$$G^n + G \to G^{n+1} ,$$

with reaction probability $P_g^+ = \gamma^+ \in [0, 1]$ with $n \geq 1$. Reciprocal catalysis of particle types A to F

$$A + B \xrightarrow{F} C ,$$

and

$$D + E \xrightarrow{C} F ,$$

with $P_r^+ = (1 + \exp[\varrho^+])^{-(1+n)^{-2}}$, $\varrho^+ \in \mathbb{R}$ and n the number of catalysts present at tile i (i.e. $n = |F_i|$ for the former reaction, and $n = |C_i|$ for the latter). For the experiments in sections 6 and 6.7, self-assembly and reciprocal catalysis are combined:

$$D + E \xrightarrow{C} F + G .$$

Containment is modeled by modifying the attachment reaction for self-assembly:

$$G^n(kC, mF) + G + pC + qF \to G^{n+1}((k + p)C, (m + q)F) .$$

The experiments in section 6.7 feature a containment capacity c, representing the maximum number of catalysts a crystal may contain:

$$G_c^n(kC, mF) + G + pC + qF \to G_c^{n+1}(k'C, m'F) + p'C + q'F ,$$

with $k' + m' \leq c$, $k' + p' = k + p$ and $m' + q' = m + q$. If $k + m + p + q > c$, the order in which catalysts are contained is determined at random.

Breakup

Particle types G, G^n, C and F have a probability of breaking up:

$$G^n \rightarrow G^{n-1} + G \,,$$

with $\gamma^- \in \mathbb{R}$ and $P_g^- = (1 + \exp[\gamma^-])^{-n}$. If the crystal contains catalysts, they are released upon detachment (irrespective of the containment capacity c):

$$G^n(kC, mF) \rightarrow G^{n-1} + G + kC + mF \,.$$

Furthermore,

$$C \rightarrow A + B \,,$$

and

$$F \rightarrow D + E \,,$$

with reaction probability $P_r^- = \varrho^- \in [0, 1]$. In sections 6 and 6.7, G particles are removed from the grid with $P_g^- = (1 + \exp[\gamma^-])^{-1}$.

PART III

CREATIVITY SUPPORT TOOLS

7

CHRONICLING CULTURAL ANCESTRY THROUGH CONCEPTUAL CLASSIFICATION

The application of phylogenetic techniques to the documentation of cultural history can present a distorted picture due to horizontal transmission (i.e. transfer to or from the lineage of a group, rather than within) and conceptual blending. Moreover, the units of cultural transmission must be communicable concepts, rather than conveniently measurable attributes, and relatedness between elements of culture often resides at the conceptual level, something not captured by phylogenetic methods, which focus on measurable attributes. For example, mortars and pestles are as related as two artifacts could be, despite little similarity at the attribute level. This chapter[1] introduces a new, cognitively inspired framework for chronicling material cultural history, building on Lipo's [110] network-based computational approach. We show that by incorporating not just superficial attributes of artifact samples (e.g. shape) but also conceptual

[1]This chapter appeared as [59] Gabora, L., Leijnen, S., Veloz, T. and Lipo, C. (2011). A Non-Phylogenetic Conceptual Network Architecture for Organizing Classes of Material Artifacts into Cultural Lineages. In *Proceedings of the Annual Meeting of the Cognitive Science Society, July 20-23, 2011, Boston, MA*:2923–2928. My contribution included implementation, experimentation and dissemination of those activities

knowledge (e.g. information about function), a different pattern of cultural ancestry emerges.

7.1 Introduction

The efforts of biologists, phylogeneticists, and others, have culminated in an impressively detailed understanding of how the living things of today evolved. We can trace the ancestral origins of our eyes and fingers, and even certain behavioral traits such as mating preferences. However, we lack comprehensive knowledge of patterns of relatedness of elements of culture, even restricting ourselves just to material artifacts.

This chapter discusses difficulties that have arisen in attempting to chronicle material cultural history using phylogenetic and network based approaches. We then describe our new conceptual network approach. The insight that guides this approach is: since artifacts are the product of minds that encode representations of them not just at the attribute level but also at an abstract, conceptual level, to reconstruct material cultural evolution, it is necessary to incorporate how artifacts are conceived, and how these conceptions interact in a human mind. We introduce a computer program that is able to construct such networks from both attribute data and conceptual information.

7.2 Phylogenetic Approaches

Since artifacts undergo 'descent with modification', the theory of natural selection appears to offer a means for explaining cultural history. Accordingly, phylogenetic methods such as cladistics are routinely borrowed from biology and applied in an archaeological context [121, 122]. In cladistic representations of archaeological data, the measured attributes of a 'taxon' of artifacts are listed as a number string. The position in the string is loosely analogous to the concept of gene, and the number at that position is loosely analogous to the concept of allele. Thus if a taxon is represented by `132` then the first attribute is in state one, the second is in state three, and the third is in state two. For example, consider the representation of early projectile points from the Southeastern United States shown in figure 7.1 [121]. The data consist of metric and morphological measurements with respect to eight attributes, each of which can take from two to six possible states. Thus, for example, if fluting is absent in a particular artifact, it has a 1 in position VII, and if fluting is present it has a 2. Seventeen 'taxa' are identified, and the pattern is such that one common ancestor (identified as KDR) gave rise to sequential branching's that culminated in 16 different taxa. This technique provides an intuitively meaningful (although potentially misleading) means of capturing structural change. The 'root taxon' at the far left is the most primitive, and early branch points represent changes that provided the structural constraints that shaped more recent changes. For example, much as evolution of the backbone paved the way for limbs, evolution of containers paved the way for spouts and handles.

Phylogenetic approaches have also been applied to culture in more complex ways. For example, relationships amongst different elements of culture have been analyzed by comparing their phylogenetic trees. The procedure involves running a series of forward models, one in which the phenomena are assumed to evolve completely independently, another in which one kind of correlation is assumed (e.g. matriliny[1] with cattle), another in which a different correlation is assumed (e.g. patriliny with cattle). These are compared to the language phylogeny, which is assumed to be the most accurate available cultural history tree, to determine which gives the best match. This method can indeed unearth relationships amongst different elements of culture. It was found, for example, that the spread of pastoralism in Sub-Saharan Africa is associated with a shift from matriliny to patriliny [81]. However, the method is ineffective if there is rampant blending of cultural elements, and it does not generate information about why or how elements of culture are related.

7.2.1 Shortcomings of Phylogenetic Approaches

Despite the intuitiveness and scientific rigor of phylogenetic/cladistic approaches, and some apparent successes applying them to culture, concerns have been raised about distortions generated by these cultural applications [55, 110, 154, 155]. We now examine these concerns.

7.2.1.1 Similarity Need Not Reflect Homology

Phylogenetic methods assume that similarity reflects homology, i.e. that two species are similar because they are related. Specifically, it assumes that, either (1) one is

[1] *matriliny* means that traits are inherited through the female line; inheritance through the male line is called *patriliny*

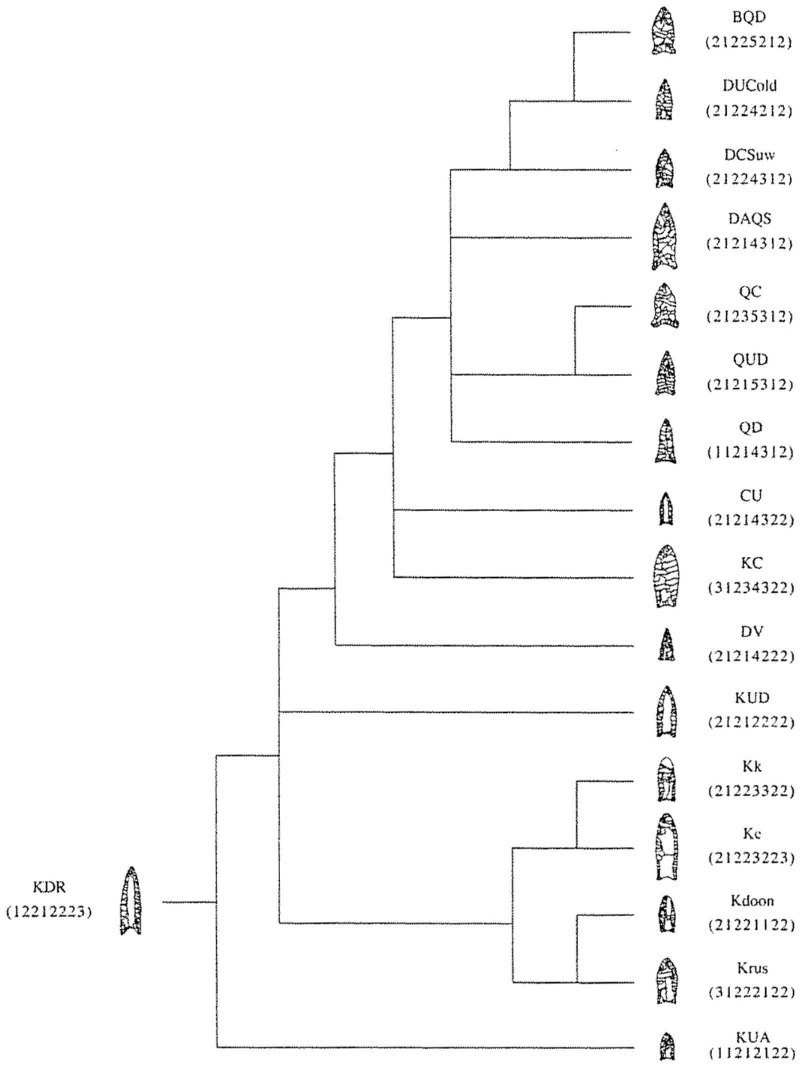


Figure 7.1: Phylogenetic representation of Paleo-Indian period projectile points from the Southeastern United States with 17 taxa defined by 18 attributes. From O'Brien et al. [121].

descended from the other, in which case shared traits were transmitted vertically, or (2) they are descended from a common ancestor, which is depicted as a branch point. For example, common ancestry can occur through fission, in which a population splits in two, which become increasingly differentiated.

However, similarity need not reflect homology. Artifacts may arise independently, yet be similar because they are alternative solutions within similar design constraints. Convergent evolution occurs in a biological context too. However, because organisms must solve many problems (reproduction, locomotion, digestion, etc.) the probability that a species is mis-categorized on the basis of how it solves any one problem is low. Artifacts, on the other hand, are generally constructed with a single use in mind. (Though artifacts developed for use in one context may be used to solve other problems, e.g., a screwdriver may be used to open a can of paint). Therefore, the probability of mis-categorization arising through the assumption that similarity reflects homology is problematic for artifacts.

7.2.1.2 Blending

Cultural relatedness frequently arises through not just vertical transmission but horizontal (inter-lineage) transmission, which can result in the blending of knowledge from different sources. Since inter-lineage transfer of information is relatively rare in animals, phylogenetic methods are ill-equipped to deal with it. Extensive horizontal transmission gives a bushy, reticulated appearance to a phylogenetic tree, which is misleading because it implies not just chronology but ancestry.

Blending is problematic for cladistic methods because it forces one to parse the data according to predefined attributes or characters. So one is a priori discouraged from incorporating data that does not fit into this parsing. In biology, such parsing arises naturally stemming from how traits are genetically encoded. The chosen attributes are characteristic of that species, and the rarity of inter-species mating ensures that they don't change drastically. However, in culture, nothing is a priori prohibited from 'mating with' anything else. Those who apply phylogenetics to culture respond that such problems rarely arise in the study of prehistory. On the basis of a set of studies of virtually indistinguishable artifacts, Collard et al. [28] insinuate that cultural blending is not widely present. This, however, reflects their highly limited choice of artifacts; a brief examination of the contents of any modern house would lead one to a different conclusion. Moreover, even if one is more interested in prehistoric culture than contemporary culture, one seeks not a bag of tricks for assessing relatedness each of which is applicable to certain data sets, but an explanatory framework that fits them all.

7.2.1.3 Lack of Objective Measure of Relatedness

A more fundamental problem with phylogenetic approaches to culture is that they assume it is possible to accurately measure the relatedness of artifacts. Whether or not two organisms share a common ancestor is clear-cut; they either are or are not descendants of a particular individual. One can objectively measure what percentage of the genomes of two species overlap, and make conclusions about their degree of genetic relatedness. But in a cultural context, whether or not two artifacts "share a common ancestor" can be arbitrary, and moreover, what is measured is not necessarily what was culturally transmitted.

7.2.1.4 Predefined Attributes

The data of figure 7.1 are typical of those to which a phylogenetic approach is amenable because the taxa are very similar to one another. That is, each taxon has one version or another of the considered attributes; there are no major modifications in this lineage. A problem pointed out by Alex Bentley (pers. com.) is that the units considered are those that are most amenable to analysis rather than those that were most likely to have been transmitted from teacher to apprentice. Thus the method documents readily measurable change, not the actual cultural ancestry of the artifact.

7.3 Lipo's Network (LN) Approach

Network-based methods appear to avert the above problems by simply ordering data according to similarity without necessarily implying common ancestry [110]. Analysis of the same data yields quite a different pattern of evolutionary change. Following O'Brien, samples that are rated the same with respect to all considered attributes are categories together as a particular taxon. Attributes are encoded as a number string. Each position in the string refers to a particular attribute, and the number at a position refers to the state of that attribute for the taxon. This is shown in figure 7.2.

Taxa are simply arranged according to the number of attributes by which they differ. The majority of taxa have two lines coming from them, one to a taxon that preceded it, and one to a taxon that followed it; the network does not specify which is which. Those that have more (e.g. `31222122`) reflect the existence of multiple other taxa with the same number of differences.

Several aspects of the procedure are noteworthy. First, the network-based approach does not make a priori assumptions about the sources of diversity. It is uncommitted with respect to whether differences reflect branching due to fission or blending due to transmission. Second, the method is also uncommitted with respect to chronology. Additional data indicate the directionality of the evolutionary pathway, as shown in figure 7.3.

7.3.1 Limitations of the LN Approach

We believe that in order to avoid the limitations of phylogenetic methods, a move in the direction of network representations is inevitable. However, this initial implementation has limitations.

7.3.1.1 Considers Only Superficial Attributes.

This approach is suitable for artifacts that are highly similar at the superficial attribute level. However, it cannot to handle artifacts whose similarity resides at the conceptual level. For example, it seems reasonable to hypothesize that the stoplight has (at least) two cultural ancestors – the streetlight and the car – the first contributing the necessary expertise (mastery over the technological design space of external lighting), and the second contributing the necessary motive (control traffic). The second is as crucial as the first; if cars (or something like them) had not come into existence, stoplights would not have come into existence. However, the network approach does not provide a way to document this. Their lack of low-level similarity means that this relationship cannot be reconstructed using this method.

7.3.1.2 Assumes Single-Attribute Change.

The LN architecture assumes that the evolutionary path cannot be resolved when there are multiple attribute differences between neighboring taxa. This is not the case when

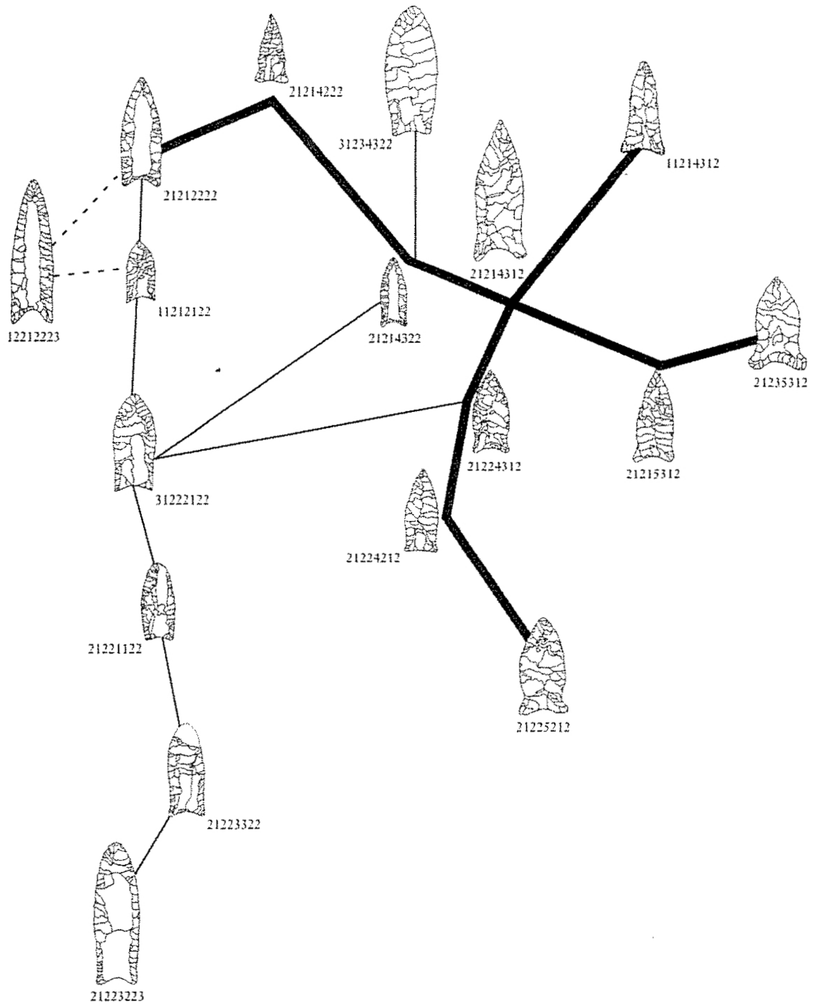


Figure 7.2: Graph produced by linking taxa to their most similar neighbors. Bold lines represent differences of only one attribute. Thin solid lines show differences of two attributes. Dotted lines show differences of three attributes. The multiple lines connecting taxon `31222122` to other taxa indicate ambiguity due to equivalent number of differences between multiple taxa. From Lipo [110].

conceptual structure is taken into account; multiple differences (or even complete lack of similarity) at the attribute level may reflect single changes at the concept level. Moreover, once the conceptual level is introduced, it is no longer necessary to restrict oneself to independent attributes. Indeed, dependencies amongst attributes may indicate the presence of conceptual structure that may hold the clue to the artifact's evolutionary story.

7.3.1.3 Constraints on Attributes.

Third, the length of the number string and the attributes considered are determined a priori according to certain rules: attributes must be independent, and there must be no significant difference in the fitness of alternative states, i.e. only neutral variation is considered. The rationale behind these rules is that they rule out similarity due to convergence (e.g. structural constraints). There is also an implied preference for data with taxa that differ from one another by only one attribute, because in such cases the pattern of ancestry can be resolved without ambiguity. When there are differences of multiple attributes between a taxon and its nearest neighbor, the evolutionary path cannot be resolved (e.g. the transition from `111` to `122` could occur by way of either `112` or `121`). The underlying assumption is that innovation involves one superficial attribute at a time, so a lack of single-attribute change between neighboring classes is assumed to indicate an incomplete data set. However, this assumption is not always met. For example, Tëmkin & Eldredge's [154] cornet data exhibits "well-documented temporally spaced sequences of "missing links" that likely indicate an actual pattern of ancestry and descent" (p. 150).

The network method has the limitation that to chronicle the evolution of a lineage that is increasing in complexity, one would either have to go backwards and add placeholders for traits that did not previously exist, or clump together a great variety of taxa as indistinguishable instances of the terminus. To document the history of human material culture, our framework must accommodate, for example, that this lineage, or one like it, eventually gave rise to the gun. The gun has few of the attributes considered thus far in analyses of this lineage such as 'fluting' or 'arc-shaped base'. Its similarity, indeed our sense that it belongs in this lineage, is conceptual; it reflects the way it is conceived of and used by humans.

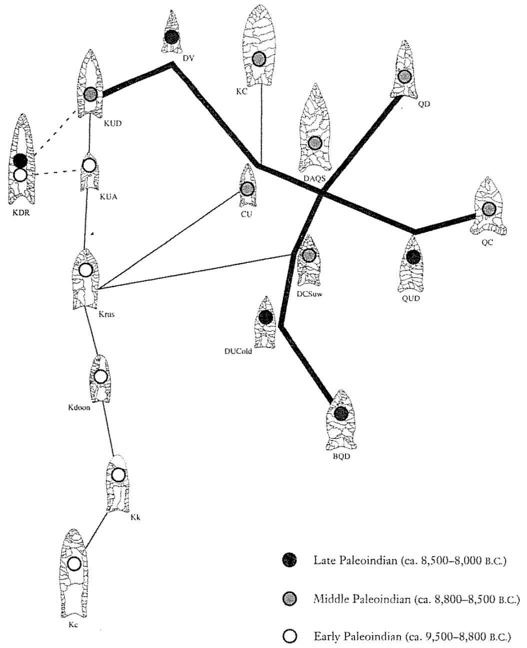


Figure 7.3: Graphical analysis of projectile point data with temporal information (from Anderson et al. [2]) indicated by degree of shading of circles. (From Lipo [110]).

In sum, the network method is a sensible, rigorous way of organizing archaeological data. However, due to its assumed independence of attributes, consideration of only superficial attributes, and fixed-length attribute strings, the resulting framework for cultural evolution is fragmentary, limited in application to what many would find the least interesting, or at any rate the least innovative, periods of cultural change.

7.4 The Conceptual Network (CN) Approach

The project described here builds on Lipo's network-based method but adds conceptual structure. As is conventional, concepts are indicated with capitals. Thus an instance of a projectile point is written as 'projectile point' but the concept of one is written as PROJECTILE POINT. The more superficial level of conceptual structure consists of what Rosch [132] refers to as basic level concepts such as PROJECTILE POINT and KNIFE. Basic level concepts mirror the attributes of objects in the external world. This basic level is the level at which items are first perceived, and it is the level at which we generally refer to and interact with them. In some cases it may be more natural to work at a finer level of discrimination and thus consider a more subordinate conceptual level, e.g. BEVELED KNIFE instead of KNIFE. The important thing is that this superficial level be rich in attributes. The less superficial, more abstract level of conceptual structure consists of superordinate concepts such as WEAPON. Superordinate concepts often refer to multiple basic level categories (e.g. PROJECTILE POINT and KNIFE are both instances of WEAPON), and they are more general than the level at which we refer to and interact with items (e.g. different kinds of weapon are interacted with in different ways). Basic level concepts and superordinate concepts can take us a long way toward a representation of how objects in the world and their interrelations are conceptualized.

To organize material culture in a way that allows for projectile points to evolve into guns, we incorporate a minimal amount of conceptual structure. The structure of the concept PROJECTILE POINT may include not just that it has certain attributes but also that it is an instance of the concept WEAPON. Sometimes the structure of concepts derives from their history (how they were conceived in the past), and sometimes from other sources (e.g. horizontal transmission or copying error). The cognitive approach uses networks to represent, not just taxa of artifacts, but relationships amongst them as they are conceived of in the mind of a particular population of individuals at a particular time and place.

The program was developed using the object-oriented Java platform with extension packages for working with networks (JUNG) and Excel files (SX). The tool collects meta-data for a set of known samples by asking the user questions about their presumed function and use. The questions are generated using a conceptual network that determines which questions are relevant for the sample. This leads to the creation of two networks: an attribute-level only one, and one that incorporates meta-data. Other software functions allow the user to export and import data sets for later use, storing meta-data and networks.

7.4.1 Data Samples

Data can be entered into the program either manually, filling out fields for each sample, or using batch excel files that contain all samples that need to be evaluated. Both import methods require a series of entry fields to be filled out in order for the program

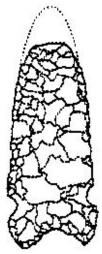


Figure 7.4: Image of Graham3 sample.

to query the user in the next stage. These entries are as follows:

Entry	Description	Example
Sample name	Unique name that identifies the sample	Graham3
Sample attributes	Features encoded as a numeric string	2262233212221
Generic type	Group to which this sample belongs	Graham Cave
Period	Estimated period of the sample's original use	7,000–5,500 BC
Location	Describes where the sample has been found	Cooper Site
Image	Picture of sample	see fig. 7.4

7.4.2 Conceptual Networks

The samples are described by a set of superficial attributes related to their relative sizes and shapes, the material from which they were constructed, and so forth. Since the intended function of an artifact does not follow unambiguously from these attributes, a human expert capable of deducing function from shape, and who may also have knowledge concerning their location and period, provides additional information to aid the computer program in determining how the samples are related. Following Dunnell [45], we define function in terms of the relationship between an object and its environment, including both natural and artificial aspects. Variability in the physical aspects of objects sometimes reflects function. For example, broad, thick objects have lower performance values than narrow ones for piercing, and objects that interact with air at any velocity are shaped by aerodynamics. Since the number of possible functions that an artifact could have is potentially infinite, the program asks only those questions that are relevant for a particular sample based on assessment of attributes. Since thus far much of the data has consisted of projectile points, all samples trigger the question, 'Was the sample a projectile point', and 'Was the sample thrown'. Other examples of questions asked include, 'Was the sample used for cutting'.

7.4.3 Database

Answers given by human experts are stored as meta-data in the program. Since for large datasets, an expert may not be able to handle the full set in one session, sets of samples may be imported from and exported to text format files.

7.4.4 Generation of Lineages

The program analyses the superficial attributes and abstract (e.g. functional) aspects of samples, and uses this information to generate networks that arrange the artifacts according to how similar they are. Thus the network shows how the artifacts are likely to have evolved chronologically. Relative distance between two samples x and y in the original network is determined by the following algorithm:

$$N(x, y) = H(f(x), f(y)) \tag{7.1}$$

where N is the distance without abstract concepts, H is the Hamming distance between two encodings and $f(x)$ is the attribute encoding of x. For the CN, the algorithm is expanded with a function over the meta-data:

$$M(x, y) = N(x, y) + D(a(x), a(y)) \tag{7.2}$$

Where D is a binary function that indicates whether two attributes are similar (0) or different (1) and $a(x)$ is a conceptual level attribute of x.

7.5 Results

Although the approach has not been tested comprehensively, in every test of ten or more samples so far there is at least one difference in the chronological ordering of between the CN approach and the original network approach. For comparative purposes, we began with the same data that was analyzed using the previously described approaches. An example of actual output of the program is given in figure 7.5. Since using the entire

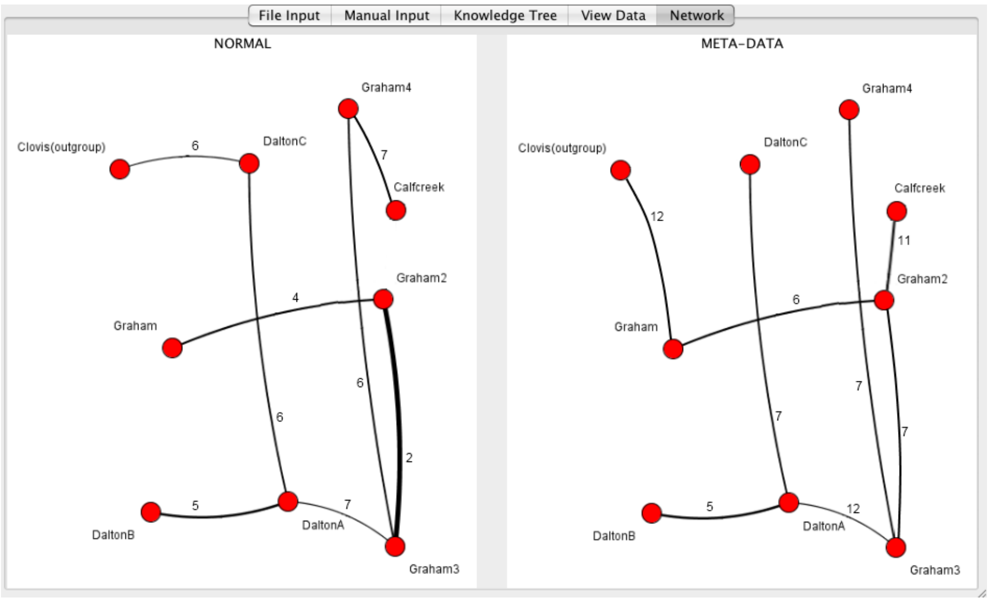


Figure 7.5: Two examples of network output given the same input data. Circles represent particular samples. Numbered lines give estimates of relatedness (lower numbers more closely related). The output on the left makes use of superficial attributes only. The output on the right additionally makes use of conceptual meta-data.

data set generates output that is crowded and difficult to parse, the figure just shows a subset of the data. The output shows both the original network approach and the CN approach. In the LN approach, shown to the left, for any sample x, it is possible that more than one of the other samples is equally similar to x, i.e. minimizes the Hamming distance (the $N(x, y)$ function) with respect to x. Therefore, using attributes only, there is a large probability of generating the incorrect lineage. Regarding the samples featured on the upper right, the method deduces that the terminal sample 'Calfcreek' is most closely related to the topmost sample, 'Graham4'. Indeed based on the superficial attributes only this was a reasonable guess. In the CN approach, however, using conceptual information (the $M(x, y)$ function), we can distinguish the correct ordering on the basis of higher-level information. We see that the LN approach guessed incorrectly, and that 'Calfcreek' is actually more closely related to 'Graham2', the one below it, than to 'Graham4', the one guessed using the LN.

7.6 Discussion

To reconstruct the history of the objects we build and use requires us to consider conceptual relationships, and indeed, to reconstruct the history of conceptual change in the minds that created them. The conceptual network approach introduced here avoids pitfalls inherent in phylogenetic approaches. It builds on an earlier network-based model, by adding the capacity to make use of not just superficial attributes of artifacts but also abstract knowledge referred to as meta-level data. Though for this initial analysis, for comparative purposes we used data that had been previously analyzed using other approaches, the current approach can readily be applied to chronicling of patterns of interrelatedness amongst artifacts of different kinds (e.g. one tool might fall into disuse when a superior tool comes into existence, or the tool for procuring a certain food might be expected to appear at the same time and location as the tool for processing it). The approach is in its infancy; we continue to improve the program through application of research from cognitive science on concept combination and the formation of hierarchical conceptual structure [27, 95]. Though preliminary, we believe that the approach holds promise in the quest to understand the ancestry of the multitude of artifacts we have created.

8

CREATIVITY IN PROCEDURAL CONTENT GENERATION FOR VIDEO GAMES

Procedural content generation (PCG) aims to algorithmically produce original solutions to game design challenges. This chapter[1] investigates how computational creativity theory can be applied to improve current PCG tools and techniques. We suggest that content generation may be considered as a dual process: a generation step to create variety and a resolution step to transform the output of the generation into a coherent and useful configuration. Separating these two steps facilitates the design of PCG algorithms and impacts the design of PCG tools.

8.1 Introduction

Procedural content generation (PCG) for games is a fast evolving discipline. Academic research constantly tries to push the boundaries of the field by improving techniques

[1]This chapter is based on [44] Dormans, J. and Leijnen, S. (2013). Combinatorial and Exploratory Creativity in Procedural Content Generation. In *Workshop Proceedings of the 8th International Conference on the Foundations of Digital Games, May 14-17, 2013, Chania, Greece*, ISBN 78-0-9913982-1-8.

based on cellular automata [88], transformational grammars [41], evolutionary algorithms [158], or answer set programming [147]. Researchers have also explored the way procedural content generation can be used to create smart, "mixed-initiative" design tools to boost developer productivity [146, 148], automate the exploration of design spaces [123] or generate rhetorical mechanics [159], to name just a few examples.

In most (if not all) applications of procedural content generation, the computer is responsible for finding solutions to design problems posed by the game's context, player's action, or designer's direction. In general, algorithms that can come up with the most original and useful solutions will be considered best, especially if the solutions are found fast and make sense within the context of the game, while they are still able to surprise both players and designers. This chapter explores what it means for an algorithm to be creative, and how a more thorough understanding of artificial creativity can guide the design of new procedural content generation algorithms and tools. The chapter sketches a new approach to procedural content generation that splits the creative process into two separate steps. Two simple experiments illustrate how this approach can be applied to procedural content generation. In the final sections, the chapter discusses how this approach might be used to measure the creativity of an algorithm, and how this approach can be used to improve automated game design tools that use procedural content generation techniques.

8.2 Designing for Creativity

Creative computational systems are sometimes modeled after a simplified evolutionary process, in which predefined sets of parameters are optimized in accordance with a selection function [82]. When these systems are used in PCG design tools, the choice of parameters and selection function becomes an optimization problem of its own: the designer of creative systems is required to balance out several features, such as

1. The size and structure of the solution space;
2. The method for generating new solutions;
3. The amount of time available for finding a sufficient solution;
4. The criteria for deeming a solution sufficient.

In computational creativity research, a distinction is often made between combinatorial creativity (combination of two previously unconnected elements) and exploratory creativity (exploration of an established conceptual space or style)[15, 120]. This paper deals with the transition from combinatorial to exploratory creativity: how can game designers develop procedural generation tools that go beyond merely permutating a set variables, while maintaining scalability and tractability?

A second useful definition is the distinction between novelty and usefulness [117]. These terms are reflected in the optimization design problem outlined above – i.e. "how are new solutions generated?" (novelty) "what constitutes a solution?" (usefulness). This suggests that the generation of novelty and the generation of usefulness might be split into two different algorithms, and that the combination of those algorithms alongside the human design process allow for optimization of design at a higher level. At this level, the particular combination of quantitative factors (e.g. algorithm execution time, computational memory, human design time) and qualitative factors (i.e. novelty, usefulness) determine whether the creative process is combinatorial or exploratory. In the approach taken here, a simple algorithm (the generation step) is responsible for generating a wide variety of data using a simple combinatorial logic. A second algorithm (the resolution step) is responsible for reorganizing the data into usable content for the game. Together, these two steps cooperate to push the creativity of the algorithm toward exploratory creativity.

Dividing the process into two steps with a clear division of responsibilities between them, has four important advantages:

1. The generation step becomes trivial to design and implement. In ideal situations any random combination will do, and in practice only a few simple constraints need to be taken into account.
2. The novelty of the produced content can be determined early on and before the computationally more expensive resolution algorithm is executed.
3. The designer of the generation algorithm only has to focus on specifying the structures that are allowed by the game. The designer does not have to specify all the possible combinations of structures, or how these combinations are to be generated.

4. The generation algorithm can be easily replaced by a different source of novelty. For example, it can be handcrafted by a designer, or produced in response to player performance.

8.3 Experiments

Previous research into procedural content generation has led to the development of Ludoscope [43]. Ludoscope is an experimental program that uses transformational grammars to generate content. It allows designers to create different types of grammars (string grammars, graph grammars, tile grammars, and shape grammars), and set up different recipes to generate content tailored toward a particular game. An important design feature of the program is its ability to split procedural content generation into multiple steps. Each step can be specified by different grammars, and produces different models that represent a game's content during various stages of its generation. Ludoscope's multi-step grammar-based operations and its relation to model driven engineering as a general approach to procedural content generation has been reported previously [41, 42].

Initially, no changes to Ludoscope were needed to set up a number of experiments to test our assumptions about simple generation of variety, combined with a more sophisticated resolution algorithm to generate a useful data set. The first experiment used a simple tile based grammar to generate dungeons that might be used in a roguelike game. The generation step simply produced a tile map randomly filled with walls and open spaces, except on the borders, which would always be walls (see figure 8.1). For this experiment, the chance that a tile (except those on the edge) was set to a wall was 40%. The resolution applies a simple transformation grammar (figure 8.3) to structure the random set into something that is more usable for a game (see figure 8.2). It is interesting to note that the results of the resolution slightly differ as the rules in the transformation grammar are applied randomly, by selecting one possible transformation from all possible transformations every step. However, they do not differ as much as one might expect from a random application of transformation rules: the grammar converges on a number of stable solutions.

In many ways, the grammar for the resolution step acts as a cellular automaton: locally defined rules, depending on a tile's neighbors, change open spaces to walls and

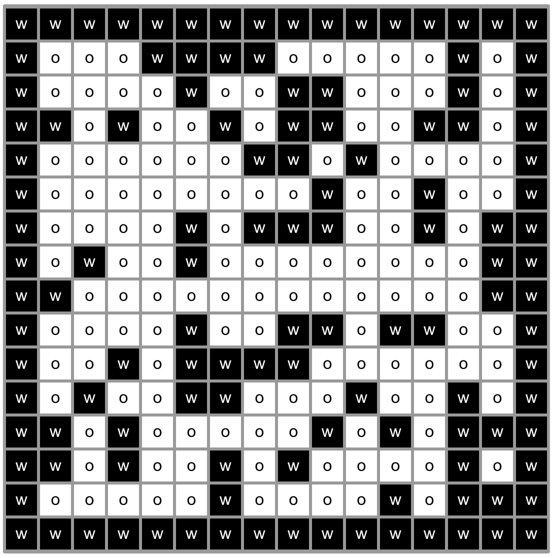


Figure 8.1: Randomly generated set of walls and open spaces to create a dungeon.

vice versa. However, it is important to note that the grammar used in the resolution step does not change the number of walls and open spaces as cellular automata would. Open spaces and walls might swap places, but their total number does not change. This feature is important when the same techniques are used to generate other elements in the dungeon at the same time. For example, when the number of monsters, doors and traps is determined using the same generation step.

The second experiment involves the generation of lock and key mission structures. In this case, the generation step produces a string of tasks. The string always starts with an "entrance" and ends with a "goal" to represent the start and end points for the level. In addition, the first task is always a key and the last task is always a lock. This guarantees that any lock is preceded by at least one key and any key is followed by at least one lock. The intermediate tasks are randomly set to contain locks, keys or other task with a ratio of 25%, 25% and 50% (see figure 8.4). Next, the resolution step generates a structure in which each key is associated to at least one lock (and vice versa), and in which the initial sequence of locks, keys and other tasks is preserved (see figures 8.5 and 8.6). Note that the grammar in figure 8.6 does not change the number or types of nodes, it only changes their connections.

An important difference between this experiment and the previous experiment is that in this case the resolution step really does produces even fewer solutions for a par-

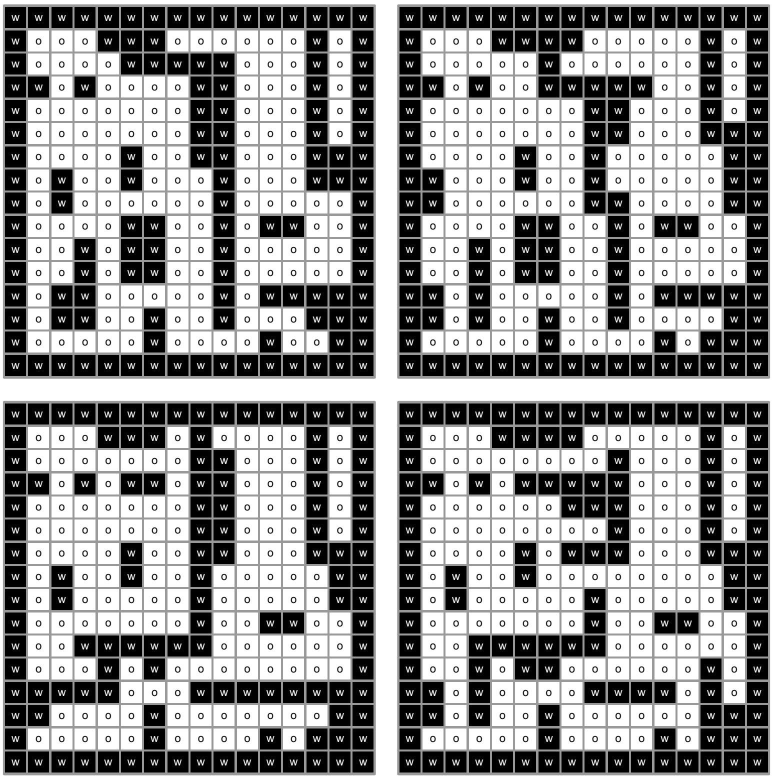


Figure 8.2: Sample results after the application an exploratory grammar to create more structure.

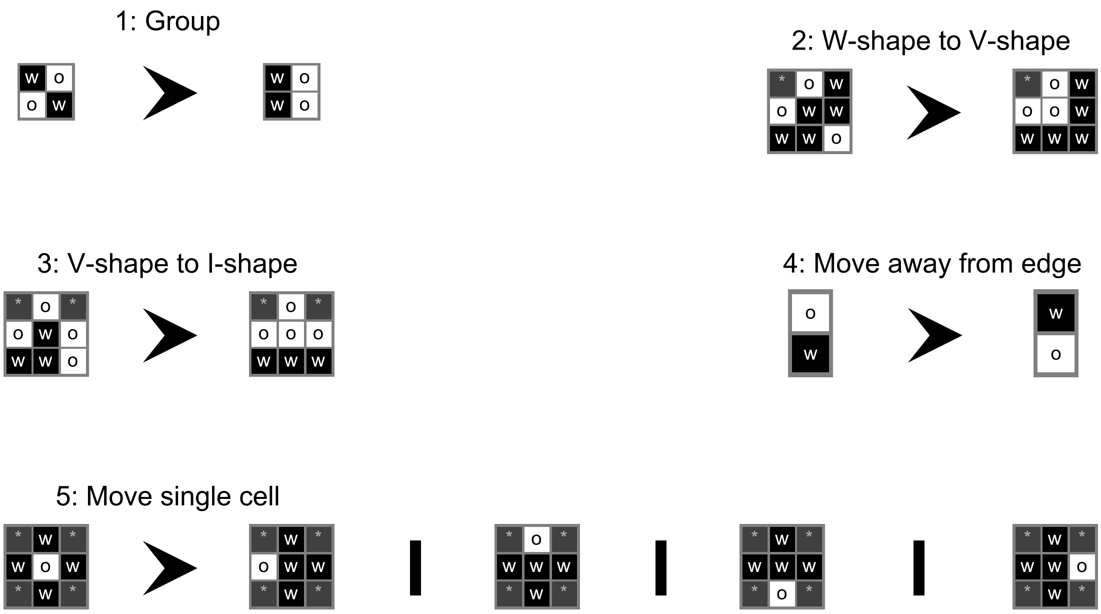


Figure 8.3: The transformation rules used to produce the results in figure 8.2 from the input in figure 8.1.

ticular generated set of locks and keys. This is guaranteed by executing each individual rule in the resolution grammar until they can no longer be applied to the mission structure. The rules are executed as they appear in figure 8.6. In many cases the generated structure is always the same; for each input there is one result. In certain cases different solutions can occur: for each iteration of the grammar execution a random applicable transformation is selected from the set of all applicable transformations. In the case of multiple keys followed by multiple locks, this leads to different possible applications of rule 5 in figure 8.6, because in that case, several keys might match node number 2 in the left-hand pattern of that rule.

A notable outcome of the second experiment is the ease with which the grammars generate a wide variety of different mission structures. The resolution grammar does not dictate how many keys are required to open a single lock, or how often it can be used to open multiple locks. Previous experiments with lock and key generation grammars required much more sophisticated grammars [41], yet did not generate the same variety of possible lock and key structures.

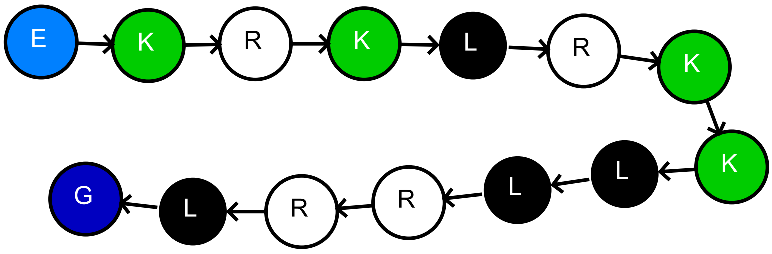


Figure 8.4: Randomly generated mission containing locks and keys (E = entrance, G = goal, L = lock, K = key, R = empty room).

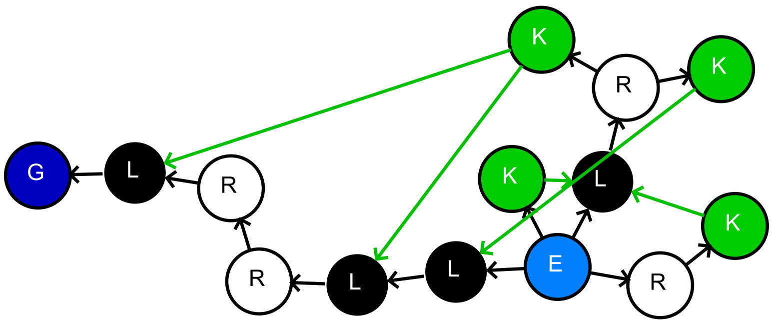


Figure 8.5: Sample result after the application a resolution grammar to create a spatial structure in which the mission of figure 8.4 is likely traversal. In this diagram, green arrows indicate which keys unlock which locks, where multiple keys might be required to unlock a single lock.

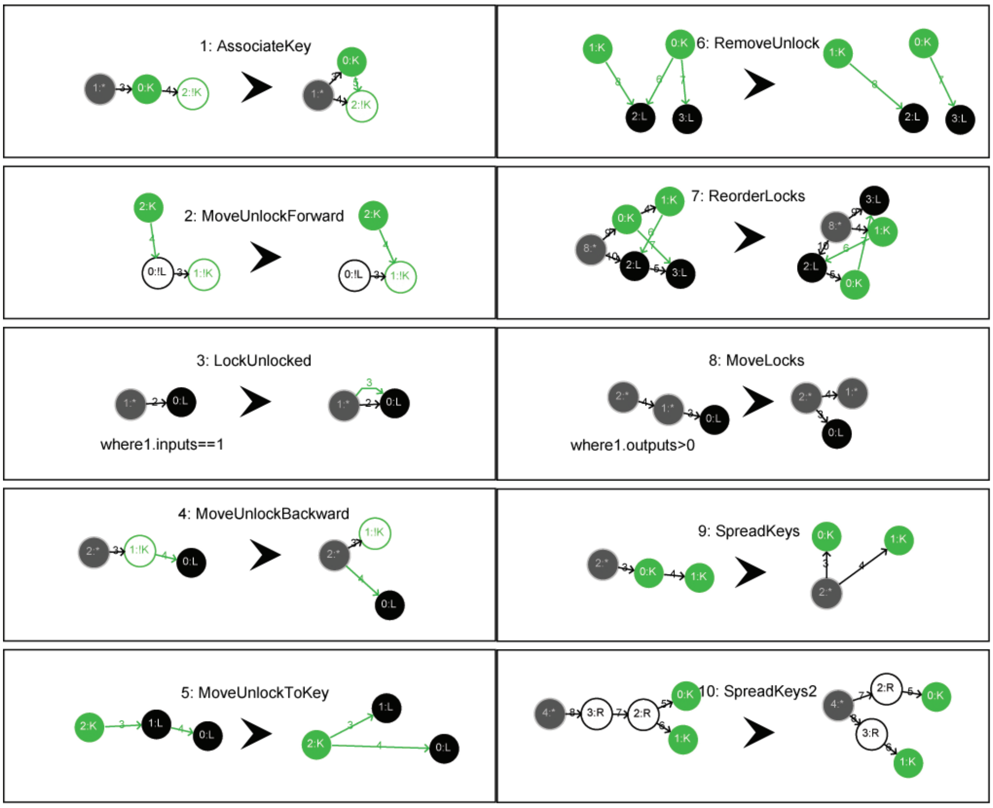


Figure 8.6: The transformation rules used to produce the results in figure 8.5.

8.4 Tailoring PCG Tools

One important result of the new approach to procedural content generation described in this chapter is a redesign of Ludoscope. Ludoscope originally was first and foremost a tool that helps designers create transformation grammars and experiment with different ways of executing multiple transformations. It was most relevant as a tool to design procedural content generation procedures for games. By explicitly supporting generation and resolution steps into this process, Ludoscope changed into a tool that fits in with "mixed-initiative" design tools [148], making it more relevant for game development as a generic content production tool.

The most important change is the creation of input and output channels in the tool's main window (see figure 8.7). A designer is able to modify the model in the input channel (in the top half). Ludoscope uses that input to create an output. The output can be the result of the execution of a single transformation grammar, or a more complex procedure specified by a recipe that is able to execute multiple grammars and apply other special operations such as converting a graph to a figure consisting of two-dimensional shapes. Depending on the speed of the transformation, the output can be generated in response to any change in the input in real time.

For example, the output generated in figure 8.7 might represent a simple level for a platform game. In this case the designer specified the location of platforms in the input channel, and the transformation grammar fills in the details to create the complete level. In this case, the designer executes the generation step manually and Ludoscope responds by filling the details; it executes the resolution step. This set-up allows a designer to explore different possibilities for distribution of platforms much faster than normally would be possible. In addition, grammars can be designed in such way that it takes into account the distance a player might be able to jump in order to make sure the level remains playable. At the moment of writing, we have only started to explore the possibilities this new approach brings.

Distinguishing between generation and the resolution steps during content generation also makes it easier to change the source of variety for the algorithm. The variety can also be deliberately designed manually, as is the case in Ludoscope. In addition, as was illustrated with the experiments above, simple random selection of elements already work if the resolution step is powerful enough to deal with (almost) all possible

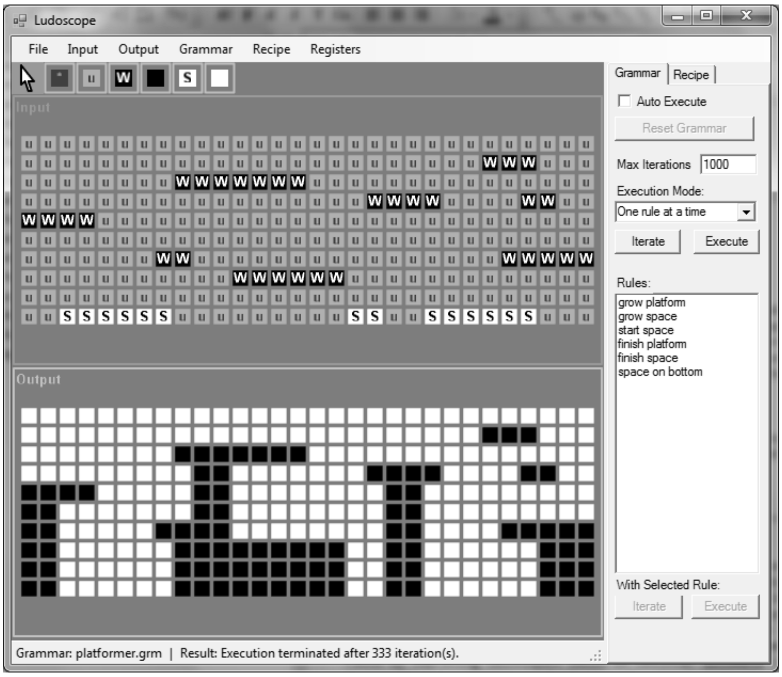


Figure 8.7: Latest version of Ludoscope featuring input and output channels.

combinations; in other words, if its divergence rate is zero or more. An option that is not explored in any detail in this chapter is to use player input to generate levels. For example, in the game *Infinite Mario* [157], the locations where players jump, pick-up coins, or defeat enemies, are recorded and used as an input to generate the next level. Whereas in the original experiment, many odd and arguably less useful levels were generated in response to player actions, the alternative of following up the player input with a resolution step might lead to more a consistent and playable game.

Furthermore, generation and resolution steps can be embedded within Ludoscope's original design philosophy of chaining transformations to break down the content generation into a multi-step, feed-forward process [42]. In this case, the output of a transformation serves as the input of the next transformation. In contrast to the previous approach, the new approach suggest that each generation step is followed by a resolution step and each resolution step is followed by a new generation step. In practice, this could mean that the first two steps generate a simple dungeon (for example the steps that where used to generate the dungeons in the first experiment) and are followed by a new generation step (for example to add traps, monsters and treasure) which in turn is followed by a new resolution step to make sure that the randomly added content is useful (for example monsters move to guard treasure, traps move to narrow passages, and so on). It is even possible that some generation steps are generated automatically, whereas others are based on user input. In the case of a process where designers can make changes to multiple steps in the process, this creates difficulties with reapplying changes that are made later in the process, over changes that are made later in time but earlier in the process. The problem is akin to allowing designers to edit multiple models that represent different perspectives on the same artifact [165]. It currently remains one of the more pressing research questions for the further development of Ludoscope.

8.5 Conclusions

Applying findings from the field of computational creativity to procedural content generation has yielded a number of interesting observations. Breaking down the content generation algorithms into a generation and resolution step facilitates the design of procedural content generation algorithms. Generation and resolution steps can be applied to a wide variety of procedural content generation techniques. In this chapter, the focus was on a number of different grammar based approaches. It is possible that cellular automata and evolutionary algorithms might equally benefit from separate generation and resolution steps. For instance, evolutionary algorithms might be applied to generate interesting inputs before they are forwarded to a resolution algorithm.

This approach offers opportunities to determine the novelty and usefulness of a generated solution early on, before the more computationally expensive resolution step is executed. This may speed up the generation procedure as a whole considerably. Ideally, the resolution step is designed not to converge or diverge any further but to stabilize the variety. This makes the resolution highly controllable and suggests new opportunities to create mixed initiative design tools. In this case, designers are responsible for creating new content while tools automatically resolve the designer's input according to the game's constraints.

EPILOGUE

Ever since he discovered how to make fire, man has stared into its flames. How often did their twisting and twirling movements remind one of a snake seizing its own tail? And how often did they bring to mind the circular structures of benzene rings? Kekulé's story is testimony of how far his mind must have diverged while dozing off, allowing him to associate flames with snakes and benzene rings. His moment of insight, however, was all but soporific: the immediate recognition of the value of the discovery is perhaps the most striking aspect of his introspective account, and also precisely what separates him from his fire-gazing predecessors.

In Part I: Creativity in an Artificial Agent Society, the relation between the conditions and consequences of historical creativity are investigated by means of a series of computer simulation experiments. The abilities of original individuals like Kekulé, the hopes, dreams and failures that so often make their stories unique and fascinating: these have all been compressed into a few variables such as "innovation rate", in order to keep the simulation computationally and programmatically tractable. However, rather than abandoning personal creativity altogether, it is merely bracketed out in the first part to make an experimental study of originality in a society of agents possible - only to resurface in the second part of this thesis.

The results described in Chapter 3 show that there exist both benefits and drawbacks to creation in a simulated agent environment. When these results were first published, they found their way to a number of internet forums where they sparked a discussion about the amount of time people should spend creating, for society to benefit most. Though an interesting topic in its own right, the conclusions drawn were beside the actual point of the paper: the findings that reported optimal creative conditions for a simulated multi-agent system cannot be directly transferred to actual groups of people, due to the limited representational scope that a simulation allows. Instead, what these findings do show is that imitation may enhance creativity for a group, as it prevents promising solutions from breaking up by acting as a memory pool, thus suggesting there may exist an ideal balance between innovation and imitation, and what conditions this balance may depend on.

Studying the profusion of ideas at the level of society allowed us to find other dynamical patterns emerging from a variety of agent interactions. Creative leadership was found to most beneficial to uncreative teams, while for creative teams, the leader should pick out the best ideas and imitate them. We also found that clustering creative agents may lead to a local amplification of good ideas and consequently a higher fitness rate over time, although it may induce the drawback of becoming trapped in a sub-optimal set of ideas. Finally, our research suggests that divergent creativity in leaders yields a higher average fitness in the short run, for instance in start-up companies; more established organizations benefit from a more conservative type of innovation.

Part II: Creativity and Constraint takes up the challenge that was bracketed out in the first part: examining and modeling creativity on the level of the individual. It deals less with originality, as it focuses instead on what kind of processes are capable of purposefully generating novelty. Chapter 4 sets out to computationally model a series of chimpanzee language learning experiments, using a genetic algorithm to train neural networks. The semiotic difference between indices and symbols forms the centerpiece of these experiments, and it is exactly this difference which cannot be expressed accurately by a non-hierarchical neural network model. Instead, the results suggested that an emergent dynamics is required for indices to point to each other in a second-order indexical (i.e. symbolic) way.

Chapter 5 provides a different but related argument against non-emergent models, this time considering the theoretical possibility of computational transformational creativity that was more broadly discussed in the introduction to this thesis. Randomness by itself is a poor substitute for autonomous purposeful novelty creation, as is deriving purpose merely from the interactions with a (human) user. As programming constitutes the generation of a specific set of constraints on a computational system, the challenge of designing self-programming systems - that require transformational creativity - begs us to consider non-computational systems capable of purposefully generating and eliminating their own constraints.

Chapter 6 connects the endings of the previous two chapters, by exploring constraint in non-computational systems in order to explain how second-order emergent dynamics may spontaneously come about. Specific conditions are necessary to generate constraints; even more specific conditions are required to simulate a model where particular constraints are preserved. Over the course of the experiments, a carefully fine-tuned environment is constructed where the particular shape of a crystal (a second-order, formal constraint) is positively or negatively correlated with its probability of continued persistence, allowing for a minimal kind of selective process, or learning.

What makes this autogenic model interesting is not the particular form that tends to persist in the simulations, but the nearly unbounded potential of an environment where a multitude of constraint generating processes take place. Similar to the encapsulation capacity of crystals, which only becomes a functional (i.e. relevant to its own persistence) dimension in the presence of a catalytic set, these constraints may interact in new dimensions that are not only unforeseen - the unknown unknown - but that also serve the purpose of propagating themselves. The Autogenic Automaton forms a stepping stone, albeit it a small one, toward an artificial creativity capable of purposefully transforming its own constraints. Further research aimed at simulating, or even physically realizing, autogenic systems may bring forth a theory of creativity that is fully grounded in the natural sciences.

Of course, in absence of a solution to this fundamental challenge we need not wait and speculate. As the final two chapters, collected in Part III: Creativity Support Tools, show, the previously discovered principles and mechanisms of creative systems may be fruitfully applied to other domains. Cultural evolution, though sharing some similarity with its biological counterpart, lacks a clearly identifiable genotype, leaving archeologists to determine what constitutes an artifact's "DNA" in order to construct a network (or tree) of cultural inheritance. Chapter 7 argues that the conceptual relationship between artifacts, present in the minds of its creators rather than in the object itself (an outlook it shares with the EVOC framework, where all innovation exists in the minds of the agents and the interaction between them) should be part of an artifact's DNA. Chapter 8 examines how techniques for automatically producing video game content may be improved by regarding them as creative systems. Alternating between a divergent algorithm that generates variety and a convergent algorithm that resolves solutions toward useful configurations facilitates mixed-initiative design, as it makes the content creation process more transparent and controllable.

The physicist Richard Feynman once proclaimed “If our small minds, for some convenience, divide this glass of wine, this universe, into parts – physics, biology, geology, astronomy, psychology, and so on – remember that nature does not know it! So let us put it all back together, not forgetting ultimately what it is for. Let it give us one more final pleasure: drink it and forget it all!”. The anthology of artificial systems presented in this thesis is intended to convey a similar quality: creativity and constraint are not as far removed from one other as the common understanding of those words might suggest.

LIST OF PUBLICATIONS

Leijnen, S. and Gabora, L. (2009). How creative should creators be to optimize the evolution of ideas? A computational model. *EPTCS, 9*:108–119.

Leijnen, S. and Gabora, L. (2009). DAWN: A Computer Model of the Development of a Creative Worldview. In *Proceedings of the 117th Annual Meeting of the American Psychological Association, August 6-9, 2009, Toronto, ON.*

Leijnen, S. and Gabora, L. (2009). The artist loft effect in the clustering of creative types. In *Creativity and Cognition 2009*:389–390.

Gabora, L. and Leijnen, S. (2009). The tradeoff between degree of creativity and number of creators in a computational model of society. In B. Cooper and V. Danos (Eds.) *Proceedings of Developments in Computational Models: Computational Models from Nature, July 11, 2009, Rhodes, Greece.*

Leijnen, S. and Gabora, L. (2010). An agent-based simulation of the effectiveness of creative leadership. In *Proceedings of the Annual Meeting of the Cognitive Science Society, August 11-14, 2010, Portland, OR*:955–960.

Gabora, L., Leijnen, S. and von Ghyczy, T. (2010). When Is Creativity Too Much of a Good Thing? A Computer Simulation. In *Proceedings of the 118th Annual Meeting of the American Psychological Association, August 12-15, 2010, San Diego, CA.*

Leijnen, S. (2011). Thinking Outside the Box: Creativity in Self-Programming Systems. *Self-Programming in AGI Systems Workshop, August 4, 2011, Mountain View, CA.*

Gabora, L., Leijnen, S., Veloz, T. and Lipo, C. (2011). A Non-Phylogenetic Conceptual Network Architecture for Organizing Classes of Material Artifacts into Cultural

Lineages. In *Proceedings of the Annual Meeting of the Cognitive Science Society, July 20-23, 2011, Boston, MA*:2923–2928.

Leijnen, S. (2012). Creative Dynamics, New Patterns in the Reservoir. *CONAS, March 29-30, 2012, Lyon, France.*

Leijnen, S. (2012). Modeling Creative Processes in Nature and Mind. *Foundations of Enactive Cognitive Science, February 27-28, 2012, Windsor, UK.*

Leijnen, S. (2012). Emerging symbols. In T. Schilhab, F. Stjernfeld and T. Deacon (Eds.) *The Symbolic Species Evolved*. Springer Biosemiotics Series, 6:253–262.

Leijnen, S. and Haselager, P. (2013). Creativity and Constraints in Self-Organizing Systems. *Conceptual and Mathematical Foundations of Embodied Intelligence, February 27-March 1, 2013, Leipzig, Germany.*

Dormans, J. and Leijnen, S. (2013). Combinatorial and Exploratory Creativity in Procedural Content Generation. In *Workshop Proceedings of the 8th International Conference on the Foundations of Digital Games, May 14-17, 2013, Chania, Greece*, ISBN 78-0-9913982-1-8.

Gabora, L., Leijnen, S. and von Ghyczy, T. (2013). Relationship between Creativity, Imitation, and Cultural Diversity. *International Journal of Software and Informatics, 7*(4):615–627.

Leijnen, S. and Dormans, J. (2014). Order of Battle: A Case Study for Designing Emergent Structure in Games. In S. Stepney and P. Andrews (Eds.) *CoSMoS 2014: Proceedings of the 7th Workshop on Complex Systems Modelling and Simulation*, Luniver Press.

Leijnen, S., Heskes, T. and Deacon, T. W. (under review). Exploring Constraint: Simulating Self-Organization and Autogenesis in the Autogenic Automaton. *Submitted to Artificial Life Journal.*

BIBLIOGRAPHY

[1] Allen, R. E. Anamnesis in Plato's "Meno and Phaedo". *The review of metaphysics* (1959), 165–174. 1

[2] Anderson, D. G., and Sassaman, K. E. *The Paleoindian and Early Archaic Southeast.* University of Alabama Press, 1996. 106

[3] Andreasen, N. Creativity and mental illness; prevalence rates in writers and their first degree relatives. *American Journal of Psychiatry 144* (1987), 1288–1292. 18

[4] Ashby, W. R. *An introduction to cybernetics.* Chapman & Hall Ltd., London, 1957. 71

[5] Basadur, M. Leading others to think innovatively together: Creative leadership. *The Leadership Quarterly 15*, 1 (2004), 103–121. 43

[6] Beer, R. D. Autopoiesis and cognition in the Game of Life. *Artificial Life 10*, 3 (2004), 309–326. 91

[7] Bellows, R. M. *Creative leadership.* Prentice-Hall, Upper Saddle River, NJ, 1959. 43

[8] Best, M. How culture can guide evolution: An inquiry into gene/meme enhancement and opposition. *Adaptive Behavior 7*, 3 (1999), 289–293. 20

[9] Best, M. Adaptive value within natural language discourse. *Interaction Studies 7*, 1 (2006), 1–15. 20

[10] Bigi, B. Using Kullback-Leibler distance for text categorization. In *Proceedings of the 25th European Conference on IR Research* (Berlin, Heidelberg, 2003), ECIR'03, Springer-Verlag, pp. 305–319. 79

[11] Bilotta, E., and Pantano, P. Emergent patterning phenomena in 2D cellular automata. *Artificial life 11*, 3 (2005), 339–362. 76

[12] Binsted, K., and Ritchie, G. An implemented model of punning riddles. In *AAAI* (1994), pp. 633–638. 6, 7

[13] Bishop, C. M. *Pattern Recognition and Machine Learning.* Springer, 2006. 56

[14] Boden, M. *The Creative Mind: Myths & Mechanisms.* Weidenfeld/Abacus & Basic Books, 1990. 3

[15] Boden, M. *The Creative Mind: Myths & Mechanisms.* Routledge, London, 2004. 3, 5, 9, 66, 115

[16] Bohm, D. J. B. *On Creativity.* Routledge, London, 1998. 66

[17] Boone, J. L., and Smith, E. A. Is it evolution yet? a critique of evolutionary archaeology. *Current Anthropology 39* (1998), 141–173. 20

[18] Bradshaw, G. F., Langley, P. W., and Simon, H. A. Studying scientific discovery by computer simulation. *American Association for the Advancement of Science* (1983). 6

[19] Bringsjord, S., Bello, P., and Ferrucci, D. Creativity, the Turing test, and the (better) Lovelace test. In *The Turing Test.* Springer, 2003, pp. 215–239. 6

[20] Buchler, J., Ed. *Philosophical Writings of Peirce.* Dover Publications, New York, 1955, ch. 7. 54

[21] Burton, W. K., and Cabrera, N. Crystal growth and surface structure. Part I. *Discuss. Faraday Soc. 5* (1949), 33–39. 80

[22] Campbell, D. T. Blind variation and selective retention in creative thought as in other knowledge processes. *Psychological Review 67* (1960), 380–400. 5

[23] Chandler, D. *Semiotics: The Basics.* Routledge, 2002. 54

[24] Clark, A. Whatever next? Predictive brains, situated agents, and the future of cognitive science. *Behav. Brain Sci* (2012), 1–86. 5

[25] Cleland, C. E. Conceptual challenges for contemporary theories of the origin(s) of life. *Current Organic Chemistry 17* (2013), 1704–1709. 70

[26] Cohen, H. The further exploits of AARON, painter. *Stanford Humanities Review 4*, 2 (1995), 141–158. 6, 7

[27] Coley, J. D., Hayes, B., Lawson, C., and Moloney, M. Knowledge, expectations, and inductive inferences within conceptual hierarchies. *Cognition 90* (2004), 217–253. 111

[28] Collard, M., Shennan, S. L., and Tehrani, J. J. Branching, blending, and the evolution of cultural similarities and differences among human populations. *Evolution and Human Behavior 27* (2006), 169–184. 101

[29] Colton, S. *Automated Theory Formation in Pure Mathematics.* Springer-Verlag, 2002. 6

[30] Colton, S. Creativity versus the perception of creativity in computational systems. In *AAAI Spring Symposium: Creative Intelligent Systems* (2008), pp. 14–20. 6

[31] Colton, S. The Painting Fool: Stories from building an automated painter. In *Computers and creativity.* Springer, 2012, pp. 3–38. 6

[32] Cover, T. M., and Thomas, J. A. *Elements of Information Theory.* Wiley, 1991. 78

[33] D. E. Rumelhart, G. E. Hinton, R. J. W. Learning representations by back-propagating errors. *Nature 323* (1986), 533–536. 22

[34] Deacon, T. W. *The Symbolic Species: the coevolution of language and the brain.* W.W. Norton and Company, New York, 1997. 59, 63

[35] Deacon, T. W. The hierarchic logic of emergence: Untangling the interdependence of evolution and self-organization. In *Evolution and Learning: The Baldwin Effect Reconsidered*, B. H. Weber and D. J. Depew, Eds. MIT Press, Cambridge, MA, 2003, pp. 273–308. 73

[36] Deacon, T. W. Universal grammar and semiotic constraints. *Language Evolution 7* (2003), 111–139. 54

[37] Deacon, T. W. Emergence: The hole at the wheel's hub. In *The Re-Emergence of Emergence*, P. Clayton and P. Davies, Eds. MIT Press, Boston, MA, 2006, pp. 111–150. 68, 73

[38] Deacon, T. W. Reciprocal linkage between self-organizing processes is sufficient for self-reproduction and evolvability. *Biological Theory 1*, 2 (2006), 136–149. 70, 74

[39] Deacon, T. W. *Incomplete Nature: How Mind Emerged from Matter.* W.W. Norton and Company, New York, NY, 2012. 70, 72

[40] Deacon, T. W., and Koutroufinis, S. A. Information, complexity and dynamic depth. *Information* (forthcoming). 91

[41] Dormans, J. Adventures in level design: Generating missions and spaces for action adventure games. In *Proceedings of the Foundations of Digital Games Conference 2010, Monterey, CA* (2010). 114, 116, 119

[42] Dormans, J. Level design as model transformation: A strategy for automated content generation. In *Proceedings of the Foundations of Digital Games Conference 2011, Bordeaux, France* (2011). 116, 123

[43] Dormans, J. *Engineering emergence: applied theory for game design.* PhD thesis, University of Amsterdam, 2012. 116

[44] Dormans, J., and Leijnen, S. Combinatorial and exploratory creativity in procedural content generation. In *Proceedings of the fourth workshop on Procedural Content Generation for Games* (2013), Foundations of Digital Games Conference. 113

[45] Dunnell, R. C. Archaeological potential of anthropological and scientific models of function. In *Archaeological Essays in Honor of Irving Benjamin Rouse*, R. C. Dunnell and E. S. Hall, Eds. Mouton, The Hague, 1978. 108

[46] Eigen, M., and Schuster, P. *The Hypercycle - A Principle of Natural Self-Organization.* Springer, Heidelberg, 1979. 72

[47] Fauconnier, G., and Turner, M. *The way we think: Conceptual blending and the mind's hidden complexities*. Basic Books, 2008. 5

[48] Fellermann, H., Rasmussen, S., Ziock, H.-J., and Solé, R. V. Life cycle of a minimal protocell—a dissipative particle dynamics study. *Artificial Life 13*, 4 (2007), 319–345. 74

[49] Feyerabend, P. Creativity: A dangerous myth. *Critical Inquiry 13* (1986), 700–711. 3

[50] Flaherty, A. W. Frontotemporal and dopaminergic control of idea generation and creative drive. *Journal of Comparative Neurology 493* (2005), 147–153. 18

[51] Fracchia, J., and Lewontin, R. C. Does culture evolve? *History and Theory 38*, 4 (1999), 52–78. 20

[52] Friston, K. The free-energy principle: a unified brain theory? *Nature Reviews Neuroscience 11*, 2 (2010), 127–138. 5

[53] Gabora, L. Meme and variations: A computer model of cultural evolution. In *Lectures in Complex Systems*, L. Nadel and D. Stein, Eds. Addison-Wesley, Reading, MA, 1995, pp. 471–486. 19, 24, 28, 33, 46

[54] Gabora, L. Ideas are not replicators but minds are. *Biology & Philosophy 19*, 1 (2004), 127–143. 20

[55] Gabora, L. The fate of evolutionary archaeology: Survival or extinction? *World Archaeology 38*, 4 (2006), 690–696. 20, 99

[56] Gabora, L. The cultural evolution of socially situated cognition. *Cognitive Systems Research 9*, 1–2 (2008), 104–113. 20

[57] Gabora, L. Evoc: A computer model of cultural evolution. In *Proceedings of the 30th Annual Meeting of the Cognitive Science Society* (2008), V. Sloutsky, B. Love, and K. McRae, Eds., Sheridan Publishing. 19, 20, 43

[58] Gabora, L. Modeling cultural dynamics. In *Adaptive Agents in a Cultural Context* (2008), Association for the Advancement of Artificial Intelligence, AAAI Press, pp. 18–25. 20, 43

[59] Gabora, L., Leijnen, S., Veloz, T., and Lipo, C. A non-phylogenetic conceptual network architecture for organizing classes of material artifacts into cultural lineages. In *Proceedings of the Annual Meeting of the Cognitive Science Society* (2011), pp. 2923–2928. 97

[60] Gabora, L., Leijnen, S., and von Ghyczy, T. When is creativity too much of a good thing? a computer simulation. In *Proceedings of the 118th Annual Meeting of the American Psychological Association* (2010). 27

[61] Gabora, L., Leijnen, S., and von Ghyczy, T. Relationship between creativity, imitation, and cultural diversity. *International Journal of Software and Informatics 7*, 4 (2013), 615–627. 27

[62] Gardner, R. A., and Gardner, B. T. Comparative psychology and language acquisition. *Psychology: the state of the art 309* (1977), 37–76. 54

[63] Gervás, P. An expert system for the composition of formal spanish poetry. *Knowledge-Based Systems 14*, 3 (2001), 181–188. 6

[64] Getling, A. V. *Rayleigh-Bénard convection: structures and dynamics*, vol. 11. World Scientific, 1998. 71

[65] Ghiseli, B. *The creative process: A symposium*, vol. 717. University of California Pr, 1952. 1

[66] Gigliotta, O., Miglino, O., and Parisi, D. Groups of agents with a leader. *Journal of Artificial Societies and Social Simulation 10*, 4 (2007). 48

[67] Gilbert, D., and Morgner, T. JFreeChart. http://www.jfree.org/jfreechart/index.html. 25

[68] Goodwin, D. W. Alcohol as muse. *American Journal of Psychotherapy 46* (1992), 422–433. 18

[69] Goodwin, D. W. *Alcohol and the Writer*. Penguin, New York, 1998. 18

[70] Goodwin, F. K., and Jamison, R. Alcohol and drug abuse in manic-depressive illness. In

Manic-depressive illness, F. K. Goodwin and K. Jamison, Eds. Oxford University Press, New York, 1990, pp. 210–226. 18

[71] Götz, I. L. On defining creativity. *The Journal of Aesthetics and Art Criticism 39*, 3 (1981), 297–301. 1

[72] Guilford, J. P. *Psychometric methods* . McGraw-Hill, 1954. 3

[73] Haken. *Information and self-organization: a macroscopic approach to complex systems.* Springer, 2006. 71

[74] Harder, M., and Polani, D. Self-organizing particle systems. *Advances in Complex Systems 16* (2013). 78

[75] Harnad, S. The symbol grounding problem. *Physica D D42* (1990), 335–346. 64

[76] Hausman, C. R. Mechanism or teleology in the creative process. *The Journal of Philosophy 58*, 20 (1961), 577–584. 4, 68

[77] Hausman, C. R. Criteria of creativity. *Philosophy and Phenomenological Research 40*, 2 (1979), 237–249. 4, 5

[78] Higgs, P. G. The mimetic transition: a simulation study of the evolution of learning by imitation. *Proceedings of the Royal Society B: Biological Sciences 267*, 1450 (2000), 1355–1361. 20

[79] Hinton, G. E., and Nowlan, S. J. How learning can guide evolution. *Complex Systems 1* (1987), 495–502. 20

[80] Hofstadter, D., and Mitchell, M. The copycat project. *Fluid concepts and creative analogies: computer models of the fundamental mechanisms of thought* (1984), 205–568. 6

[81] Holden, C. J., and Mace, R. Spread of cattle led to the loss of matriliny in africa: a coevolutionary analysis. *Proceedings of the Royal Society B: Biological Sciences 270*, 1532 (2003), 2425–2433. 99

[82] Holland, J. *Adaptation in Natural and Artificial Systems.* MIT Press, Cambridge, MA, 1992. 57, 114

[83] Hookway, C. *Peirce.* Routledge & Kegan Paul, London, 1985. 54

[84] Hutchins, E., and Hazelhurst, B. Learning in the cultural process. In *Artificial Life II*, C. Langton, J. Taylor, D. Farmer, and S. Rasmussen, Eds. Addison-Wesley, 1991. 20

[85] Jamieson, K. R. Mood disorders and patterns of creativity in writers and artists. *Psychiatry 52* (1989), 125–134. 18

[86] Jamieson, K. R. *Touched by fire: Manic-depressive illness and the artistic temperament.* Free Press, New York, 1993. 18

[87] Jeffreys, M. The meme metaphor. *Perspectives in Biology and Medicine 43*, 2 (2000), 227–242. 20

[88] Johnson, L., Yannakakis, G. N., and Togelius, J. Cellular automata for real-time generation of infinite cave levels. In *Proceedings of the Foundations of Digital Games Conference 2010, Monterey, CA* (2010). 114

[89] Jost, L. Entropy and diversity. *Oikos 113* (2006), 363–375. 78

[90] Joyce, G. F. RNA evolution and the origins of life. *Nature 338* (1989), 217–224. 70

[91] Kant, I. The Critique of Judgment: II. Teleological Judgment. In *The Great Books*, trans. James Creed Meredith, Ed., vol. 42. University of Chicago Press, Chicago, IL, 1790/1952, pp. 550–613. 13

[92] Kauffman, S. A. *Origins of Order: self-organization and selection in evolution.* Oxford University Press, New York, NY, 1993. 78

[93] Kauffman, S. A. Approaches to the origin of life on earth. *Life 1*, 1 (2011), 34–48. 70

[94] Kaufman, J. C., and Sternberg, R. J. *The international handbook of creativity.* Cambridge University Press, 2006. 3

[95] Kemp, C., and Tenenbaum, J. B. The discovery of structural form. *Proceedings of the National Academy of Sciences 105*, 31 (2008), 10687–10692. 111

[96] Koestler, A. *The Act of Creation.* Penguin Books, New York, 1964. 5, 68

[97] KOHONEN, T. Self-organized formation of topologically correct feature maps. *Biological Cybernetics 43* (1982), 59–69. 64

[98] KRASNOGOR, N., AND GUSTAFSON, S. A study on the use of "self-generation" in memetic algorithms. *Natural Computing 3*, 1 (2004), 53–76. 19

[99] KRONFELDNER, M. E. Creativity naturalized. *The Philosophical Quarterly 237* (2009), 577–592. 4, 67

[100] KULLBACK, S., AND LEIBLER, R. On information and sufficiency. *The Annals of Mathematical Statistics 22*, 1 (1951), 79–86. 78

[101] LADY ADA LOVELACE. Note G, 1843. 6

[102] LANGTON, C. G. Studying Artificial Life with Cellular Automata. *Physica D 22* (1986), 120–149. 70

[103] LEIJNEN, S. Thinking outside the box: Creativity in self-programming systems. In *Proceedings of the Workshop on Self-Programming in AGI Systems* (Mountain View, CA, 2011). 65

[104] LEIJNEN, S. Emerging symbols. In *The Symbolic Species Evolved*, T. S. Schilhab, F. Stjernfelt, and T. W. Deacon, Eds., vol. 6. Springer, Dordrecht, the Netherlands, 2012, pp. 253–262. 53

[105] LEIJNEN, S., AND GABORA, L. The artist loft effect in the clustering of creative types. In *Creativity & Cognition 2009* (New York, NY, 2009), ACM, pp. 389–390. 27

[106] LEIJNEN, S., AND GABORA, L. How creative should creators be to optimize the evolution of ideas? a computational model. *EPTCS 9* (2009), 108–119. 17, 27

[107] LEIJNEN, S., AND GABORA, L. What is the optimal proportion of 'creative types' in society? a computer simulation. In *Creativity & Cognition 2009* (2009), ACM. 19, 27

[108] LEIJNEN, S., AND GABORA, L. An agent-based simulation of the effectiveness of creative leadership. In *Proceedings of the Annual Meeting of the Cognitive Science Society* (2010), pp. 955–960. 27

[109] LEIJNEN, S., HESKES, T., AND DEACON, T. W. Exploring constraint: Simulating self-organization and autogenesis in the Autogenic Automaton. *Artificial Life* (under review). 69

[110] LIPO, C. P. The resolution of cultural phylogenies using graphs. In *Mapping our Ancestors*, C. P. Lipo, M. J. O'Brien, M. Collard, and S. J. Shena, Eds. Transaction Publishers, New Brunswick, 2005. 97, 99, 102, 104, 106

[111] LOYD, S. *Sam Loyd's Cyclopedia of 5,000 Puzzles, Tricks, and Conundrums*. Pinnacle Books, 1914. 8

[112] LUDWIG, A. M. *The Price of Greatness*. Guilford Press, New York, 1995. 18

[113] MATURANA, H. R., AND VARELA, F. J. *Autopoiesis and cognition*. Reidel, Boston, MA, 1980. 70, 74

[114] MCCARTHY, J., MINSKY, M. L., ROCHESTER, N., AND SHANNON, C. E. A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. *AI Magazine 27*, 4 (2006), 12. 6

[115] MCMULLIN, B., AND VARELA, F. J. Rediscovering computational autopoiesis. In *Fourth European Conference on Artificial Life*, P. Husbands and I. Harvey, Eds. MIT Press, Cambridge, MA, 1997, pp. 38–48. 76

[116] MITRA, D. Computational creativity: Three generations of research and beyond. In *AAAI Spring Symposium: Creative Intelligent Systems* (2008), pp. 47–52. 6

[117] NEWELL, A., SHAW, J. G., AND SIMON, H. A. The processes of creative thinking. In *Contemporary Approaches to Creative Thinking*, H. E. Gruber, G. Terrell, and M. Wertheimer, Eds. Atherton, New York, NY, 1963, pp. 63–119. 115

[118] NORLANDER, T., AND GUSTAFSON, R. Effects of alcohol on scientific thought during the incubation phase of the creative process. *Journal of Creative Behavior 30* (1996), 231–248. 18

[119] NORLANDER, T., AND GUSTAFSON, R. Effects of alcohol on a divergent figural fluency test during the illumination phase of the creative process. *Creativity Research Journal 11* (1998), 365–374. 18

[120] Novitz, D. Creativity and constraint. *Australasian Journal of Philosophy 77* (1999), 67–82. 5, 115

[121] O'Brien, M. J., Darwent, J., and Lyman, R. L. Cladistics is useful for reconstructing archaeological phylogenies: Paleoindian points from the southeastern united states. *Journal of Archaeological Science 28* (2001), 1115–1136. 98, 100

[122] O'Brien, M. J., Lyman, R. L., Saab, Y., Saab, E., Darwent, J., and Glover, D. S. Two issues in archaeological hylogenetics: Taxon construction and outgroup selection. *Journal of Theoretical Biology 215* (2002), 133–150. 98

[123] Pantaleev, A. In search of patterns: Disrupting RPG classes through procedural content generation. In *Proceedings of the Foundations of Digital Games Conference 2012, Raleigh, NC* (2012). 114

[124] Peirce, C. S. *The Essential Peirce: Selected Philosophical Writings*, vol. 2. Indiana University Press, 1998, ch. What Is a Sign? 64

[125] Plasson, R., Brandenburg, A., Jullien, L., and Bersini, H. Autocatalysis: At the root of self-replication. *Artificial Life 17*, 3 (2011), 219–236. 70

[126] Plato, and Sharples, R. *Plato: Meno.* Aris and Phillips, 1985. 8

[127] Polani, D. Foundations and formalizations of self-organization. *Advances in Applied Self-Organizing Systems* (2008), 19–37. 78

[128] Premack, D. *Intelligence in ape and man.* Lawrence Erlbaum Associates, New Jersey, 1976. 54

[129] Prigogine, I., and Stengers, I. *Order Out of Chaos.* Bantam Books, New York, NY, 1984. 70, 71

[130] Puccio, G. J., Murdock, M., and Mance, M. *Creative leadership: skills that drive change.* Thousand Oaks, Sage, CA, 2006. 43

[131] Richards, R. E. *Everyday creativity and new views of human nature: Psychological, social, and spiritual perspectives.* American Psychological Association, 2007. 3

[132] Rosch, E. H., Mervis, C. B., Gray, W. D., Johnson, D. M., and Boyes-Braem, P. Basic objects in natural categories. *Cognitive Psychology 8*, 3 (1976), 382–439. 106

[133] Rosen, R. *Life itself: A Comprehensive Inquiry into the Nature, Origin, and Fabrication of Life.* Columbia University Press, New York, 1991. 70

[134] Rothenberg, A. Creativity, mental health, and alcoholism. *Creativity Research Journal 3* (1990), 179–201. 18

[135] Rothenberg, A. Creative cognitive processes in kekulé's discovery of the structure of the benzene molecule. *The American Journal of Psychology 108*, 3 (1995), 419–438. 2

[136] Rumbaugh, D. *Language Learning by a Chimpanzee: The Lana Project.* Academic Press, New York, 1977. 54

[137] Savage-Rumbaugh, S. Symbolization, language and chimpanzees: A theoretical reevaluation on initial language acquisition processes in four young pan troglodytes. *Brain and Language 6* (1978), 265–300. 53, 54, 55

[138] Schmidhuber, J. Formal theory of creativity, fun, and intrinsic motivation (1990–2010). *Autonomous Mental Development, IEEE Transactions on 2*, 3 (2010), 230–247. 5

[139] Searle, J. R., et al. Minds, brains, and programs. *Behavioral and brain sciences 3*, 3 (1980), 417–457. 5

[140] Seely, H. *Pieces of intelligence: The existential poetry of Donald H. Rumsfeld.* Simonand-Schuster. com, 2010. 9

[141] Shannon, C. E. A mathematical theory of communication. *Bell Systems Technical Journal 27* (1948), 379–423. 78

[142] Simon, H. A. What we know about the creative process. In *Frontiers in creativity and innovative management*, R. Kuhn, Ed. Ballinger, Cambridge, MA, 1986. 43

[143] Simonton, D. K. *Genius, creativity, and leadership: Historiometric inquiries.* Harvard University Press, Cambridge, MA, 1984. 3

[144] Simonton, D. K. *Scientific genius: A psychology of science.* Cambridge University Press, Cambridge, MA, 1988. 3

[145] Simonton, D. K. Creativity as blind variation and selective retention: Is the creative process Darwinian? *Psychological Inquiry 10* (1999), 309–328. 5

[146] Smelik, R., Turenel, T., de Kraker, K. J., and Bidarra, R. Inegrating procedural generation and manual editing of virtual worlds. In *Proceedings of the Foundations of Digital Games Conference 2010, Monterey, CA* (2010). 114

[147] Smith, A. M., and Mateas, M. Answer set programming for procedural content generation: A design space approach. *Computational Intelligence and AI in Games, IEEE Transactions on 3*, 3 (sept. 2011), 187 –200. 114

[148] Smith, G., Whitehead, J., and Mateas, M. Tanagra: A mixed-initiative level design tool. In *Proceedings of the Foundations of Digital Games Conference 2010, Monterey, CA* (2010), pp. 209–216. 114, 122

[149] Sternberg, R. J., Kaufman, J. C., and Pretz, J. E. A propulsion model of creative leadership. *The Leadership Quarterly 14*, 4–5 (2003), 455–473. 43

[150] Sternberg, R. J., and Lubart, T. I. *Defying the Crowd: Cultivating Creativity in a Culture of Conformity.* Free Press, New York, 1995. 18

[151] Sternberg, R. J., and Lubart, T. I. The concept of creativity: Prospects and paradigms. *Handbook of creativity 1* (1999), 3–15. 4, 5

[152] Styron, W. *Darkness Visible: A Memoir of Madness.* Random House, New York, 1990. 18

[153] Sulloway, F. *Born to Rebel.* Pantheon, New York, 1996. 18

[154] T emkin, I., and Eldredge, N. Phylogenetics and material cultural evolution. *Current Anthropology 48* (2007), 146–153. 99, 105

[155] Terrell, J. E., Hunt, T. L., and Gosden, C. The dimensions of social life in the pacific: Human diversity and the myth of the primitive isolate. *Current Anthropology 38* (1997), 155–195. 99

[156] Thompson, E. *Mind in Life.* Harvard University Press, Cambridge, MA, 2010. 70

[157] Togelius, J., Kastbjerg, E., Schedl, D., and Yannakakis, G. N. What is procedural content generation? Mario on the borderline. In *Proceedings of the Foundations of Digital Games Conference 2011, Bordeaux, France* (2011). 123

[158] Togelius, J., Preuss, M., and Yannakakis, G. N. Towards multiobjective procedural map generation. In *Proceedings of the Foundations of Digital Games Conference 2010, Monterey, CA* (2010). 114

[159] Treanor, M., Blackford, B., Mateas, M., and Bogost, I. Game-O-Matic: Generating videogames that represent ideas. In *Proceedings of the Foundations of Digital Games Conference 2012, Raleigh, NC* (2012). 114

[160] Turing, A. M. Computing machinery and intelligence. *Mind 59*, 236 (1950), 433–460. 5, 6

[161] Tyson, J. L. *The Belousov-Zhabotinski Reaction.* Springer-Verlag, Berlin, 1976. 71

[162] Varela, F. J., Maturana, H. R., and Uribe, R. Autopoiesis: The organization of living systems, its characterization and a model. *BioSystems 5* (1974), 187–196. 76

[163] von Neumann, J. *The Theory of Self-Reproducing Automata.* University of Illinois Press, Chicago, IL, 1966. 70

[164] Wallas, G. *The art of thought.* London, 1926. 2

[165] Walter, R., and Neuwald, K. Calliope-d: An authoring environment for interactive dialog. In *Mensch & Computer 2012: interaktiv informiert – allgegenwärtig und allumfassend!?* (München, 2012), H. Reiterer and O. Deussen, Eds., Oldenbourg Verlag, pp. 409–418. 124

[166] Wiggins, G. A. A preliminary framework for description, analysis and comparison of creative systems. *Knowledge-Based Systems 19*, 7 (2006), 449–458. 6

[167] Wiggins, G. A., Pearce, M. T., and Müllensiefen, D. Computational modelling of music cognition and musical creativity. *Oxford handbook of computer music* (2009), 383–420. 6

CURRICULUM VITAE

After spending his first year at the University of Amsterdam studying Psychology and Computer Science, Stefan Leijnen moved to Utrecht University in his second year, where he obtained a Master's degree in Artificial Intelligence in 2008. During his studies, he worked for several months at the University of Liverpool, doing research on emotion in artificial agents. He also co-created the Dutch AIBO team that participated in several Robocup world-cup tournaments, as well as launching a mobile technology startup. He did his Master's thesis project at the University of California, Berkeley, where he created a neural network model for symbolic learning, under the supervision of Terrence Deacon, Marco Wiering and John-Jules Meyer. In 2009, he started his PhD at the University of British Columbia, Kelowna, where he worked on multi-agent models of cultural evolution. In the same year, he followed his supervisor Liane Gabora to the PACE center at Tufts University, Boston, carrying on his investigation of the mechanisms underlying creativity. In 2011, he continued his PhD research at the Intelligent Systems Group at the Radboud University Nijmegen under the supervision of Tom Heskes, while making several short visits to Deacon's group in Berkeley. He also launched two new startups, Leijnen Technology and AI Labs, and works as lecturer and researcher at the Game Development department of the Amsterdam University of Applied Sciences.

SAMENVATTING

Sinds de opkomst van de computer en het internet in het dagelijks leven zijn we meer dan ooit omgeven met systemen die intelligent gedrag vertonen. Intelligent gedrag impliceert echter nog geen creativiteit: computers zijn vaak geprogrammeerd met een bepaalde uitkomst in gedachten. Ook wanneer deze uitkomst niet vooraf vaststaat, zoals bij het voorspellen van het weer of het vertalen van gebruikersinvoer, zijn de mogelijke uitkomsten vooraf vastgesteld door de ontwerper.

Zijn ontworpen systemen in staat hun eigen uitkomsten te creëren en daarnaar te handelen - met andere woorden, kunnen ze creatief zijn? De acht hoofdstukken in dit proefschrift, gepresenteerd in drie afzonderlijke delen, gaan dieper in op deze vraag. In het eerste deel wordt een samenleving van creatieve individuen nagebootst die af en toe nieuwe ideeën bedenken, maar die vaak ook de ideeën van anderen in hun omgeving imiteren. In deze simulaties wordt gezocht naar de optimale condities voor het ontstaan van goede ideeën, hoe de diversiteit hiervan kan worden bevorderd, en hoe de creativiteit van een leider een organisatie kan beïnvloeden.

Het centrale deel van dit proefschrift gaat in op de vraag of kunstmatige systemen buiten hun eigen kaders kunnen treden. Na een theoretische analyse van de systeemvereisten voor creativiteit, volgend op een hoofdstuk over de simulatie van taalbegrip met neurale netwerken, wordt in het afsluitende hoofdstuk van dit deel een biochemisch model gepresenteerd voor het ontstaan van een systeem dat zijn eigen kaders creërt en elimineert – een noodzakelijke en voldoende voorwaarde voor transformatieve creativiteit.

In het derde en laatste deel wordt tenslotte ingegaan op een tweetal toepassingen die creatieve processen ondersteunen: een applicatie voor het ordenen van pijlpunten op basis van culturele overerving, en een duaal model voor het genereren van de inhoud van computerspellen.

Zodoende worden in dit proefschrift vele variaties op het centrale thema van kunstmatige creativiteit behandeld: van innovatie, imitatie, taalbegrip, de rol van beperking, en het ontstaan van leven, tot aan archeologie en computerspellen.

SUMMARY

Since the advent of computers and the internet in everyday life, we are more than ever surrounded by systems that exhibit intelligent behavior. However, intelligent behavior doesn't necessarily imply creativity: computers are often programmed to achieve specific results. When the results are not explicitly specified, the space of possible outcomes is implicitly present in the program's design.

Can artificial systems be capable of creating new possibilities - in other words, can they be creative? In the eight chapters of this thesis, presented in three separate parts, this question is explored.

The first part features an artificial society of creative agents that occasionally invent new ideas, but will often also imitate other agents in their surroundings. Using computer simulation, the ideal conditions for the development of good ideas are investigated, as well as how the diversity of ideas can be improved, and how creative leadership may affect an organization.

The central part of this thesis deals with the question of whether artificial systems can transcend their own constraints. After a theoretical analysis of the systemic requirements for creativity, following a chapter on simulating language understanding using neural networks, the final chapter of this part presents a biochemical model for the emergence of a system that creates and eliminates its own constraints – a necessary and sufficient condition for transformational creativity.

In the third and final part, two applications that support creative processes are presented: a computer program for automatic classification of arrowheads basis on cultural inheritance, and a dual model for generating video game content.

This thesis displays a wide variety of topics, ranging from innovation and imitation, to language understanding in chimpanzees, to the relevance of constraints to creativity, to the origins of life, to archeology and computer games; each chapter representing a variation on the central theme of artificial creativity.